Business Case: Aerofit - Descriptive Statistics & Probability

Introduction

Aerofit is a leading brand in the field of fitness equipment. Aerofit provides a product range including machines such as treadmills, exercise bikes, gym equipment, and fitness accessories to cater to the needs of all categories of people.

Business Problem:

The market research team at AeroFit wants to identify the characteristics of the target audience for each type of treadmill offered by the company, to provide a better recommendation of the treadmills to the new customers. The team decides to investigate whether there are differences across the product with respect to customer characteristics.

In [3]:
```
#importing all libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from scipy import stats
```

In [4]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [5]:
```
#read the Aerofit dataset csv file

data = pd.read_csv("https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/001/125/original/aerofit_treadmill.csv?16399
```

In [6]:
```
#showing first five records of dataset
data.head()
```

Out[6]:

	Product	Age	Gender	Education	MaritalStatus	Usage	Fitness	Income	Miles
0	KP281	18	Male	14	Single	3	4	29562	112
1	KP281	19	Male	15	Single	2	3	31836	75
2	KP281	19	Female	14	Partnered	4	3	30699	66
3	KP281	19	Male	12	Single	3	3	32973	85
4	KP281	20	Male	13	Partnered	4	2	35247	47

In [7]:
```
#basic information about dataset
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 180 entries, 0 to 179
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Product        180 non-null    object
 1   Age            180 non-null    int64 
 2   Gender         180 non-null    object
 3   Education      180 non-null    int64 
 4   MaritalStatus  180 non-null    object
 5   Usage          180 non-null    int64 
 6   Fitness        180 non-null    int64 
 7   Income         180 non-null    int64 
 8   Miles          180 non-null    int64 
dtypes: int64(6), object(3)
memory usage: 12.8+ KB
```

1. The columns in dataset are "Product","Age","Gender","Education","MaritalStatus","Usage", "Fitness","Income" and "Miles
2. "Product","Gender" and "MaritalStatus" are of Object datatype and contains categorical data.
3. Remaining columns contain integer or numeric data.

```
4. There are 180 entries for each coloumn.
```

Q1

Defining Problem Statement and Analysing basic metrics:

```
As stated by Aerofit, the purpose, vision and mission of them are as follows:
1. Purpose : To be a leader in the fitness industry by providing enhanced services, Customer relationship
   and unique products.
2. Vision : To provide quality services that exceeds the expectations of our esteemed customers.
3. Mission : Statement To built long term relationship with our customers and clients and provide exceptional
   customers services by pursuing business through innovation and advances technology.
```

```
    Problem Statement:
    Aerofit company imports fitness equipment with better technology to carter to the ever expanding
    markets of commercial and domestic fitness equipment sector.
    In order to compete with all the other multinational brands, it needs to ramp up by giving similar
    technology products but at a much reasonable price.
    Company needs to analyse customer base so that three different category of Treadmill products can be
    sell at faster rate with maximum quantity.
```

```
    Basic metrics:

    1. Know Your Customer:
       Customers are the pillars of any business. It is very important to know about your customer,
       their likes and dislikes. Some prefer basic fitness while some customers are fitness freak.
       On each criteria, it is important to conclude that on which product category, customers are
       inclined.

    2. Increase Customer base:
       Customer base is one of the most important metric. It directly impacts the business revenue and
       the position hold in the market. Aerofit needs to attract more customers. As customer base
       increases, data will be more and hence more information leads to better insights and therefore
       can help in taking better decisions.

    3. Customer retention:
       While attracting new customers, it is also important to retain the existing customers. The data
       for existing customers leads to some insights which in turn can increase addition of new customers.

    4. Popularity of product category:
       It is important to know the popularity of product category for different customer base.
```

5. Customer Enagagement in Product:
   It is important to know how much customer is giving time to company's product. If the customer is using product once a week, there will be no increase in demand of that product. It is necessary to know, how much customer is engaging with the company's product.

6. Revenue and Marketing:
   It is important to understand how much revenue has been generated from various category products. Metrics such as average revenue per customer and lifetime customer value are too important for any business. They can drive the business into new directions and can help in taking better decisions for pricing and marketing strategy to get maximum revenue.

Q1.1

Observations on shape of data

In [8]:
```
#Observations on shape of data
data.shape
print(f"Number of rows: {data.shape[0]}\nNumber of columns: {data.shape[1]}")
```

```
Number of rows: 180
Number of columns: 9
```

1. The dataset contains 180 records of three products along with their 9 attributes.
2. In simpler terms, there are 180 rows and 9 columns.

data types of all the attributes

In [9]:
```
# data types of all the attributes
data.dtypes
```

Out[9]:
```
Product          object
Age               int64
Gender           object
Education         int64
MaritalStatus    object
Usage             int64
Fitness           int64
Income            int64
Miles             int64
dtype: object
```

1. There are total 9 attributes in dataset.
2. The attributes "Product", "Gender" and "MaritalStatus" are of object type attribute.
3. Remaining attributes are of integer type attributes.

In [10]:
```
#conversion of categorical attributes to 'category' (If required)
#not required
```

statistical summary

In [11]:
```
#statistical summary of columns containing categorical data
data.describe(include = object).T
```

Out[11]:

	count	unique	top	freq
Product	180	3	KP281	80
Gender	180	2	Male	104
MaritalStatus	180	2	Partnered	107

1. There are total 180 counts.
2. There are 3 unique Products.
3. The Gender column has been categorized into 2 categories.

4. The Marital Status has been categorized into 2 categories.

In [12]:
```
#statistical summary of columns containing numerical data
data.describe()
```

Out[12]:

	Age	Education	Usage	Fitness	Income	Miles
count	180.000000	180.000000	180.000000	180.000000	180.000000	180.000000
mean	28.788889	15.572222	3.455556	3.311111	53719.577778	103.194444
std	6.943498	1.617055	1.084797	0.958869	16506.684226	51.863605
min	18.000000	12.000000	2.000000	1.000000	29562.000000	21.000000
25%	24.000000	14.000000	3.000000	3.000000	44058.750000	66.000000
50%	26.000000	16.000000	3.000000	3.000000	50596.500000	94.000000
75%	33.000000	16.000000	4.000000	4.000000	58668.000000	114.750000
max	50.000000	21.000000	7.000000	5.000000	104581.000000	360.000000

1. There are total 180 counts for each column.

Age
1. The mean age of customers is around 28 years. Hence, it can be inferred that youth are more inclined towards their fitness.
2. The minimum age of customer is 18 years and maximum age of customer is 50 years.
3. 25% of customers's age are below 24 years.
4. 50% of customers's age are below 26 years.
5. 75% of customers's age are below 33 years.

Education
1. The average years of customers spend in education is around 15 years. Here, it can be inferred that customers are more likely to be educated.
2. The minimum and maximum year spend by customer in education are 12 years and 21 years respectively.
3. 25% of customers's are having experience in education below 14 years.
4. 50% of customers's are having experience in education below 16 years.
5. 75% of customers's are having experience in education below 16 years.

Usage
1. The average number of times the customer plans to use the treadmill each week is approximately 2 days.
2. The minimum and maximum average day that customer use treadmill is 2 days and 7 days respectively.
3. 25% of customers's use treadmill on average 3 days a week.
4. 50% of customers's use treadmill on average 3 days a week.
5. 75% of customers's use treadmill on average 4 days a week.

Fitness
1. On scale from 1 to 5, the average self-rated fitness is around 3.33.
2. The minimum and maximum self-rating that customer has used is 1 and 5 respectively.
3. 25% of customers's has self rated themselves under 3.
4. 50% of customers's has self rated themselves under 3.
5. 75% of customers's has self rated themselves under 4.

Income
1. The average annual income of customers is 53719.57 dollars.
2. The minimum and maximum annual income of customer is 16506.68 dollars and 104581.00 dollars respectively.
3. 25% of customers have annual income below 44058.75 dollars.
4. 50% of customers have annual income below 50596.50 dollars.
5. 75% of customers have annual income below 58668.00 dollars.
6. Standard deviation for Income is very high. This attribute might have outliers in it.

Miles
1. The average number of miles the customer expects to walk/run each week is approximately 103.2 miles.
2. The minimum and maximum average miles that customer walk/run each week is 21 miles and 360 miles respectively.
3. 25% of customers on average walk/run 66 miles a week.
4. 25% of customers on average walk/run 66 miles a week.
5. 25% of customers on average walk/run 66 miles a week.
6. Standard deviation for Miles is very high. This might have outliers in it.

In [13]:
```
#Analysis of missing values
data.isna().sum()

#It can be observed that none of the columns of dataset have missing values.
```

Out[13]:
```
Product          0
Age              0
Gender           0
Education        0
MaritalStatus    0
Usage            0
Fitness          0
Income           0
Miles            0
dtype: int64
```

Q2

Non-Graphical Analysis: Value counts and unique attributes

In [14]:
```
#Analysis data of Product column
d1 = pd.DataFrame(data["Product"].value_counts().reset_index())
d1.columns = ["Product Category","Count"]
d1.insert(2, "% count", [round(data["Product"].value_counts()[0]/len(data)*100,2),
                         round(data["Product"].value_counts()[1]/len(data)*100,2),
                         round(data["Product"].value_counts()[2]/len(data)*100,2)], True)
display(d1)
```

	Product Category	Count	% count
0	KP281	80	44.44
1	KP481	60	33.33
2	KP781	40	22.22

```
1. There are three types of treadmill product in dataset, named as "KP281","KP481" and "KP781".
2. The product KP281 was using by 80 customers. Hence, KP281 is the most frequent product.
3. The product KP481 was using by 60 customers.
4. The product KP781 was using by 40 customers.
5. The highest percentage is 44.44% for product KP281.
```

In [15]:
```
#Analysis data of Gender column
d2 =pd.DataFrame(data["Gender"].value_counts().reset_index())
d2.columns = ["Gender","Count"]
d2.insert(2, "% count", [round(data["Gender"].value_counts()[0]/len(data)*100,2),
                         round(data["Gender"].value_counts()[1]/len(data)*100,2)], True)
display(d2)
```

	Gender	Count	% count
0	Male	104	57.78
1	Female	76	42.22

```
1. There are two categories of Gender specified in dataset as "Male" and "Female".
2. There are total 104 Male customers and 76 Female customers.
```

3. 57.78% customers are male and 42.22% are females.

In [16]:
```
#Analysis data of Marital Status column
d3 = pd.DataFrame(data[["MaritalStatus"]].value_counts().reset_index())
d3.columns = ["Marital Status","Count"]
d3.insert(2, "% count", [round(data["MaritalStatus"].value_counts()[0]/len(data)*100,2),
                         round(data["MaritalStatus"].value_counts()[1]/len(data)*100,2)], True)
display(d3)
```

	Marital Status	Count	% count
0	Partnered	107	59.44
1	Single	73	40.56

1. The customers in dataset are either "single" or "Partnered".
2. There are total 73 Singles and 107 Partnered customers.
3. 59.44% customers are partnered and 40.56% customers are single.

In [17]:
```
#Analysis data of Usage column
d4 = pd.DataFrame(data["Usage"].value_counts().reset_index())
d4.columns = ["Usage","Count"]
d4.insert(2, "% count", [round(d4["Count"][0]/sum(d4["Count"])*100,2),
                         round(d4["Count"][1]/sum(d4["Count"])*100,2),
                         round(d4["Count"][2]/sum(d4["Count"])*100,2),
                         round(d4["Count"][3]/sum(d4["Count"])*100,2),
                         round(d4["Count"][4]/sum(d4["Count"])*100,2),
                         round(d4["Count"][5]/sum(d4["Count"])*100,2)], True)
display(d4)
```

	Usage	Count	% count
0	3	69	38.33
1	4	52	28.89
2	2	33	18.33
3	5	17	9.44
4	6	7	3.89
5	7	2	1.11

1. Usage field refers to average number of times the customer plans to use the treadmill each week.
2. It can be observed that about 38% of customers are using treadmill 3 days a week.
3. Only 1% of customers use whole week.
4. Customers use the product mostly 3 and 4 days a week on average.

In [18]:
```
#Analysis data of Fitness column
d5 = pd.DataFrame(data["Fitness"].value_counts().reset_index())
d5.columns = ["Fitness","Count"]
d5.insert(2, "% count", [round(d5["Count"][0]/sum(d5["Count"])*100,2),
                         round(d5["Count"][1]/sum(d5["Count"])*100,2),
                         round(d5["Count"][2]/sum(d5["Count"])*100,2),
                         round(d5["Count"][3]/sum(d5["Count"])*100,2),
                         round(d5["Count"][4]/sum(d5["Count"])*100,2)], True)
display(d5)
```

	Fitness	Count	% count
0	3	97	53.89
1	5	31	17.22
2	2	26	14.44
3	4	24	13.33
4	1	2	1.11

1. Fitness field refers to self rated finess customer has provided.
2. Rating of 1 refers to poor shape whereas 5 signifies excellent shape.
3. Only 1% customers feel that their fitness is in poor shape.
4. About 17% customers consider their fitness in excellent shape.
5. Most of customers have rated 3, that is, they consider their fitness neither in excellent shape nor in poor shape.

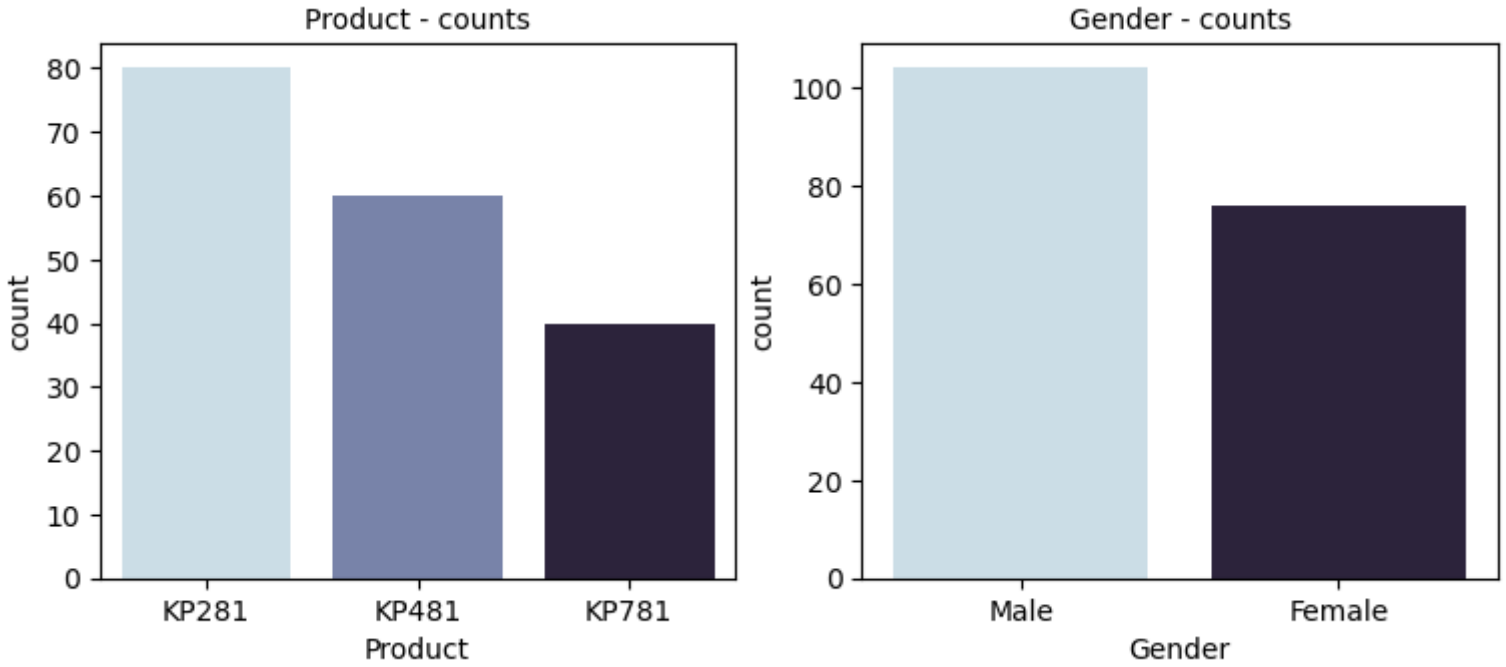
In [19]:
```
#Analysis data of Education column
d6 = pd.DataFrame(data["Education"].value_counts().reset_index())
d6.columns = ["Education","Count"]
d6.insert(2, "% count", [round(d6["Count"][0]/sum(d6["Count"])*100,2),
                         round(d6["Count"][1]/sum(d6["Count"])*100,2),
                         round(d6["Count"][2]/sum(d6["Count"])*100,2),
                         round(d6["Count"][3]/sum(d6["Count"])*100,2),
                         round(d6["Count"][4]/sum(d6["Count"])*100,2),
                         round(d6["Count"][5]/sum(d6["Count"])*100,2),
                         round(d6["Count"][6]/sum(d6["Count"])*100,2),
                         round(d6["Count"][7]/sum(d6["Count"])*100,2)], True)
display(d6)
```

	Education	Count	% count
0	16	85	47.22
1	14	55	30.56
2	18	23	12.78
3	15	5	2.78
4	13	5	2.78
5	12	3	1.67
6	21	3	1.67
7	20	1	0.56

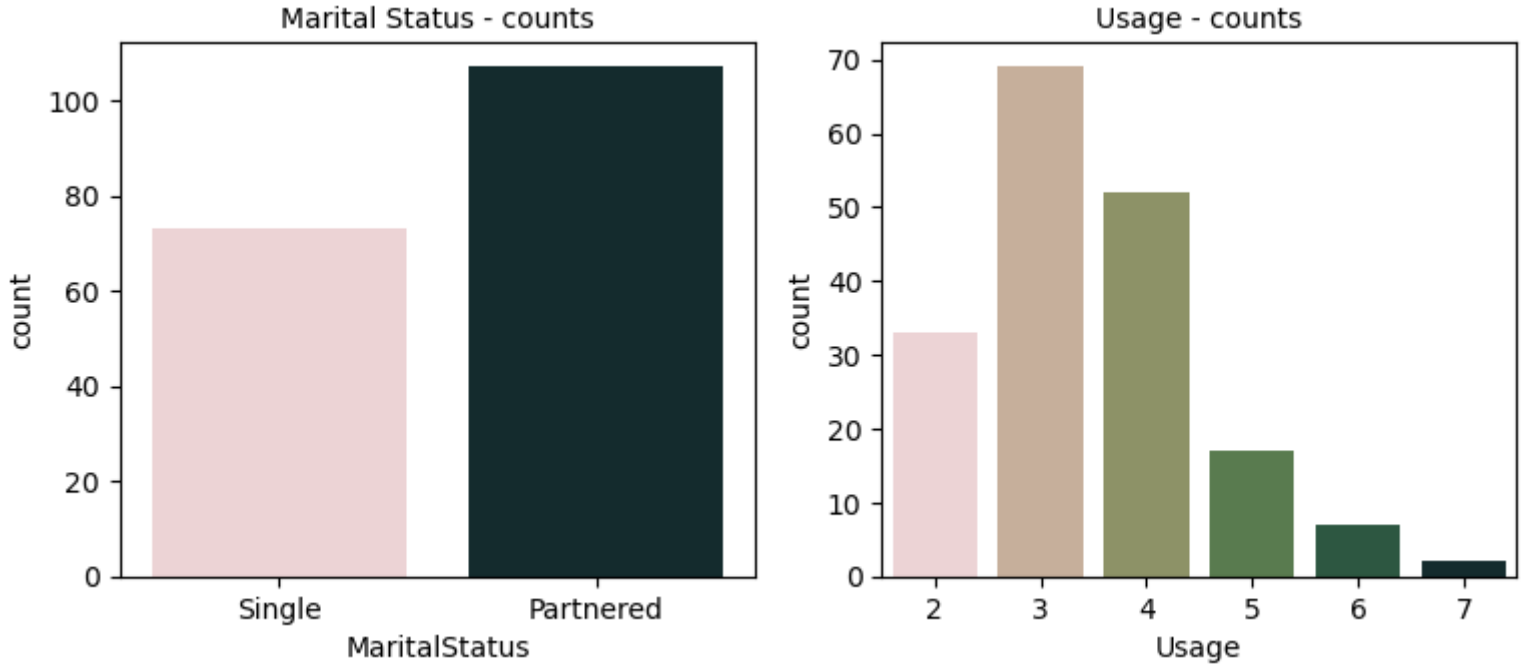
1. Education field represents the number of years customer spent in their education.
2. The percentage for minimum number of education years(12 years) and maximum number of education years(21 years) is 1.67%. This infers that most of the customers are neither highly educated nor under-educated.
3. The maximum number of customers(47.22%) have spent 16 years in education, followed by 30.56% customers spent 14 years in education.

UNIVARIATE ANALYSIS

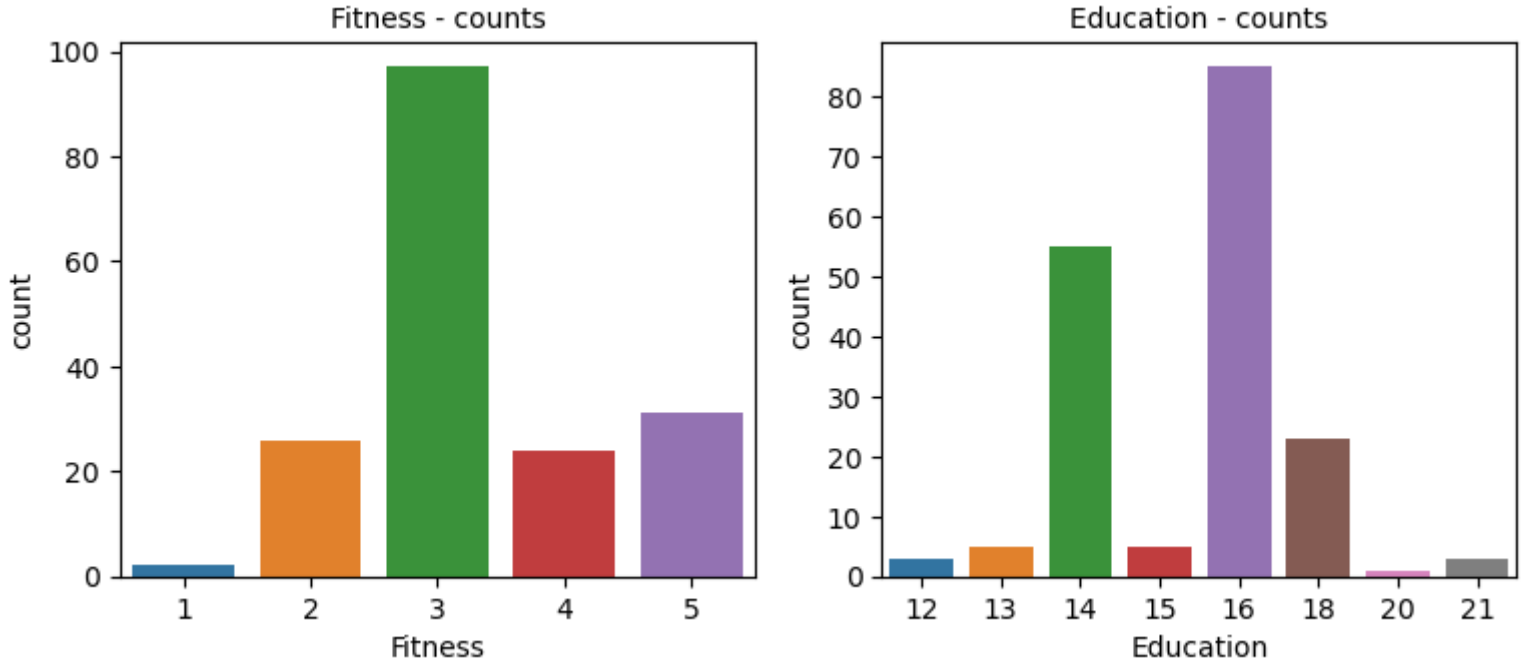
In [85]:
```
#Graphs/Plots for product, Gender, Marital Status, Usage, Fitness and Education
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.countplot(data["Product"],palette='ch:s=.25,rot=-.25',ax=axis[0])
axis[0].set_title("Product - counts", pad=5, fontsize=10)
sns.countplot(data["Gender"],palette='ch:s=.25,rot=-.25', ax=axis[1])
axis[1].set_title("Gender - counts", pad=5, fontsize=10)
plt.show()
```



In [80]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.countplot(data["MaritalStatus"],palette='ch:s=-.2,rot=-.75', ax=axis[0])
axis[0].set_title("Marital Status - counts", pad=5, fontsize=10)
sns.countplot(data["Usage"],palette='ch:s=-.2,rot=-.75',ax=axis[1])
axis[1].set_title("Usage - counts", pad=5, fontsize=10)
plt.show()
```



In [81]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.countplot(data["Fitness"], ax=axis[0])
axis[0].set_title("Fitness - counts", pad=5, fontsize=10)
sns.countplot(data["Education"],ax=axis[1])
axis[1].set_title("Education - counts", pad=5, fontsize=10)
plt.show()
```

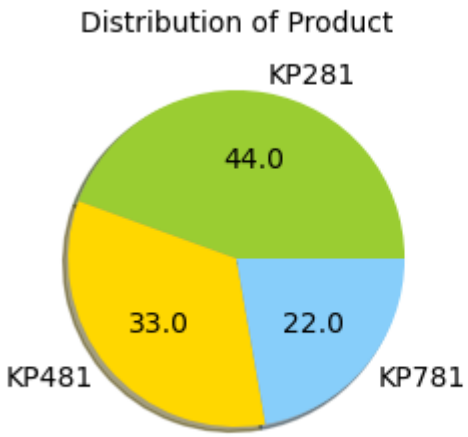


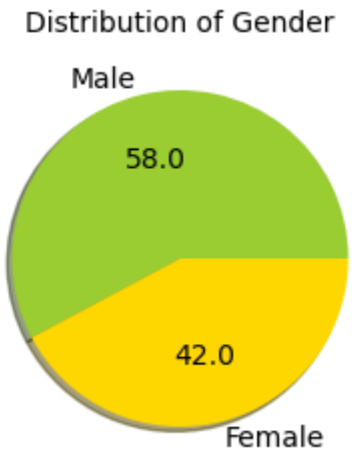
From the above plots
1. Customers are likely to use product KP281, followed by KP481. Very less customers are interested in using product KP781.
2. Males are likely to use treadmill in comparison with females.
3. Customers having partner are likely to use treadmill in comparison with Single customers.
4. Customers are likely to use product on average of 3 days a week.
5. Most of the customers have rated themselves as 3 in category of self-rated fitness. This concludes that most of the customers considered themselves as neither in excellent shape nor in poor shape.
6. Most of the customers have spent 14 to 18 years in education.

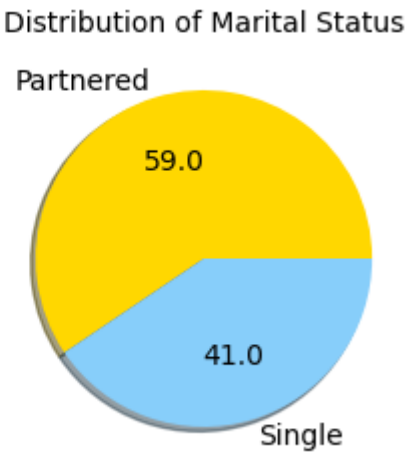
Distribution of all categorical fields in percentage

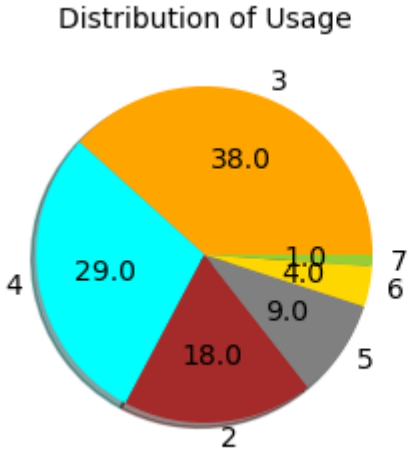
In [21]:
```
plt.figure(figsize = (10,6))

plt.subplot(2,3,1)
colors = ['yellowgreen','gold', 'lightskyblue']
def absolute_value(val):
    a  = np.round(val/100.*d1["% count"].sum(), 0)
    return a
plt.pie(d1["% count"], labels = d1["Product Category"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Product", fontsize = 10)

plt.subplot(2,3,2)
colors = ['yellowgreen','gold']
def absolute_value(val):
    a  = np.round(val/100.*d2["% count"].sum(), 0)
    return a
plt.pie(d2["% count"], labels = d2["Gender"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Gender", fontsize = 10)

plt.subplot(2,3,3)
colors = ['gold', 'lightskyblue']
def absolute_value(val):
    a  = np.round(val/100.*d3["% count"].sum(), 0)
    return a
plt.pie(d3["% count"], labels = d3["Marital Status"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Marital Status", fontsize = 10)

plt.subplot(2,3,4)
colors = ["orange", "cyan", "brown","grey", 'gold', 'yellowgreen']
def absolute_value(val):
    a  = np.round(val/100.*d4["% count"].sum(), 0)
    return a
plt.pie(d4["% count"], labels = d4["Usage"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Usage", fontsize = 10)

plt.subplot(2,3,5)
colors = ["cyan", "brown","grey", 'gold', 'yellowgreen']
def absolute_value(val):
    a  = np.round(val/100.*d5["% count"].sum(), 0)
    return a
plt.pie(d5["% count"], labels = d5["Fitness"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Fitness", fontsize = 10)

plt.subplot(2,3,6)
colors = ["orange", "cyan", "brown","grey", "beige", 'yellowgreen','gold', 'lightskyblue']
def absolute_value(val):
    a  = np.round(val/100.*d6["% count"].sum(), 0)
```

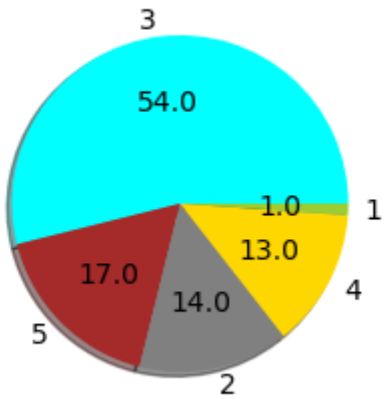
```
    return a
plt.pie(d6["% count"], labels = d6["Education"],colors=colors, autopct=absolute_value, shadow=True)
plt.title("Distribution of Education", fontsize = 10)

plt.show()

#Above information has been shown in pie chart for better clarity in percentages.
```

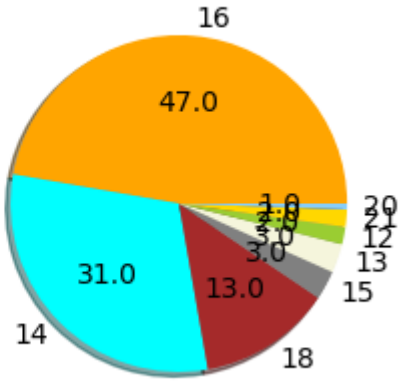




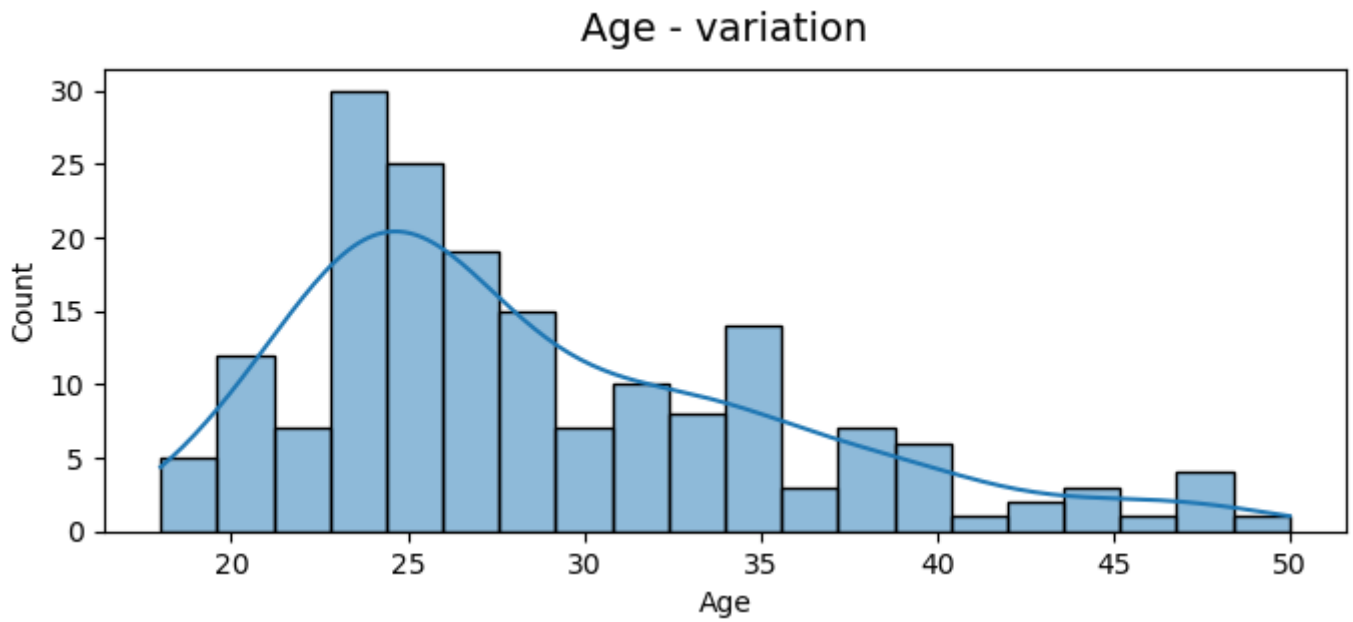




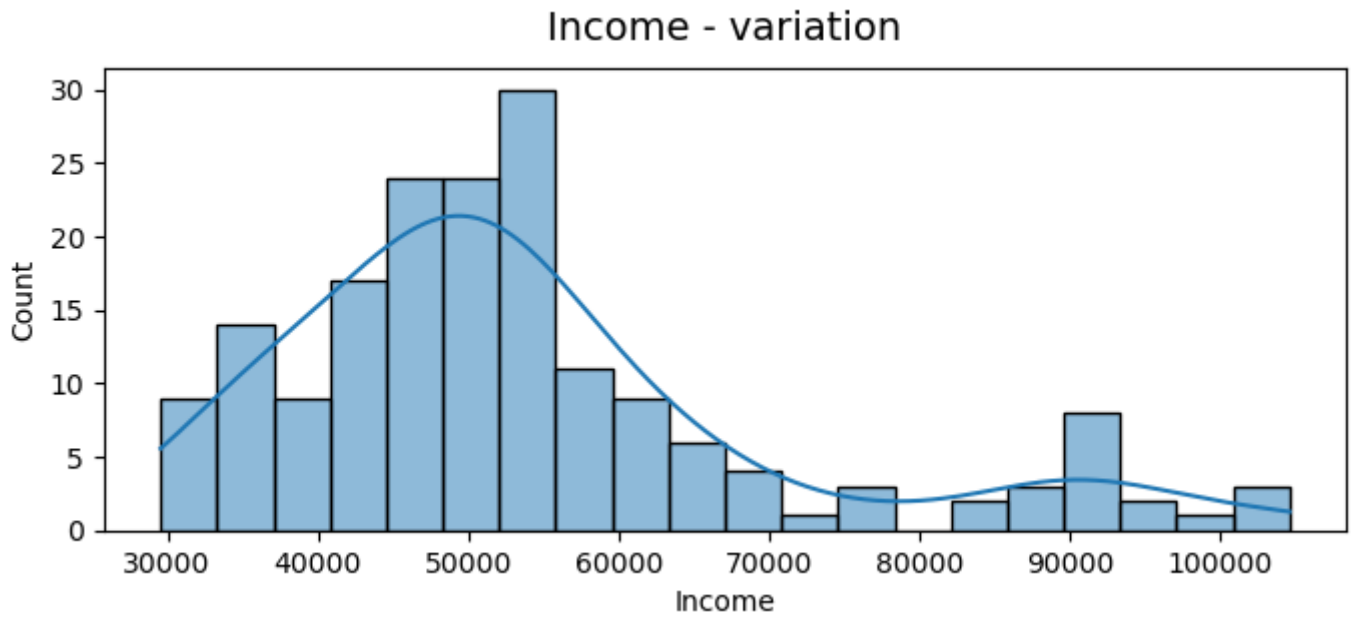




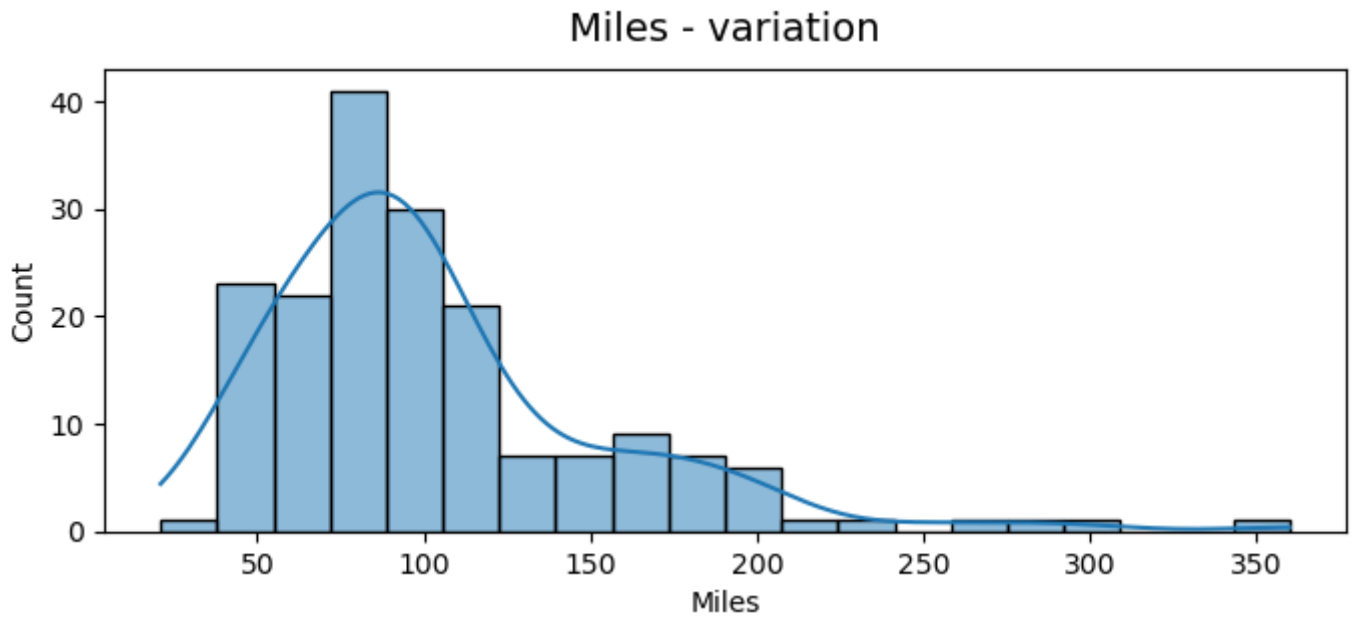
In [22]:
```
#Graphs/Plots for Age
plt.figure(figsize = (8,3))
sns.histplot(data["Age"],bins = 20, kde = True)
plt.title("Age - variation", pad=10, fontsize=14)
plt.show()

# Most of the customers using treadmill have age around 25 years.
# Very less customers are in bracket of 40 to 50 years of age.
```



In [23]:
```
#Graphs/Plots for Income
plt.figure(figsize = (8,3))
sns.histplot(data["Income"],bins = 20, kde = True)
plt.title("Income - variation", pad=10, fontsize=14)
plt.show()

# Most of the customers in dataset have income in bracket of 45k to 55k.
# Very less customers have income around 80k.
```



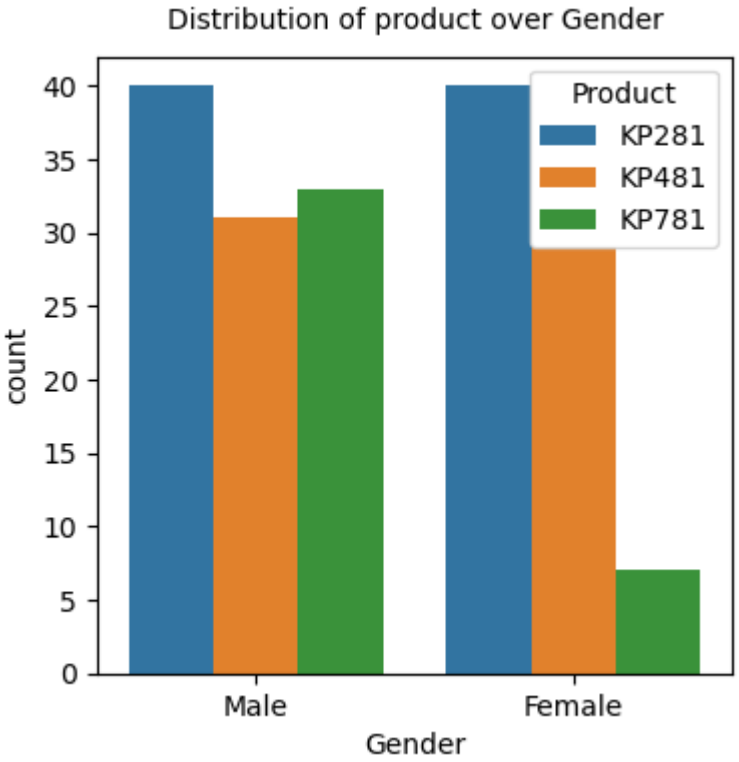
In [24]:
```
#Graphs/Plots for Miles
plt.figure(figsize = (8,3))
sns.histplot(data["Miles"],bins = 20, kde = True)
plt.title("Miles - variation", pad=10, fontsize=14)
plt.show()

# Most of the customers in dataset covers 75 to 80 miles in a week.
# Very less customers are able to cover more than 200 miles in a week.
```

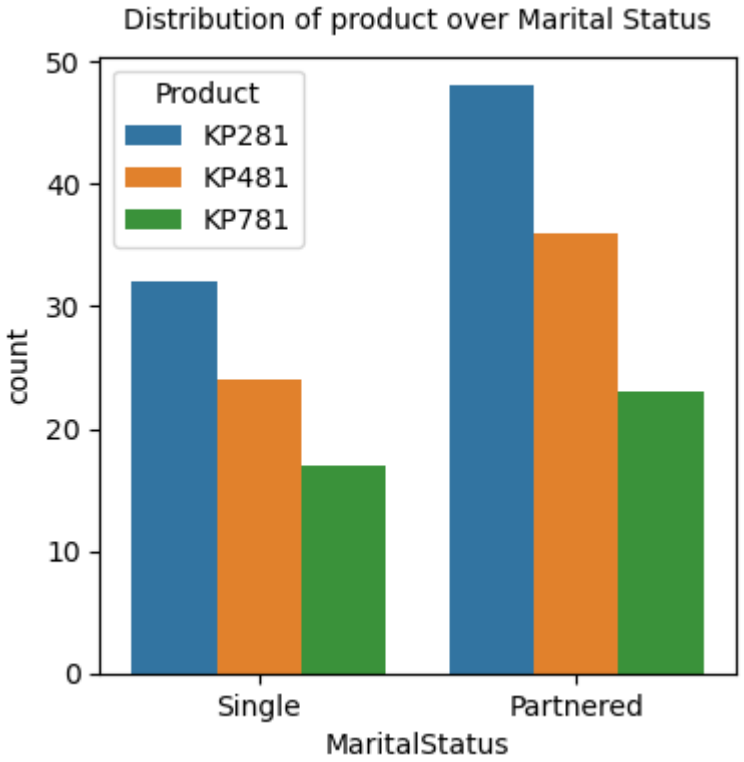


BIVARIATE ANALYSIS

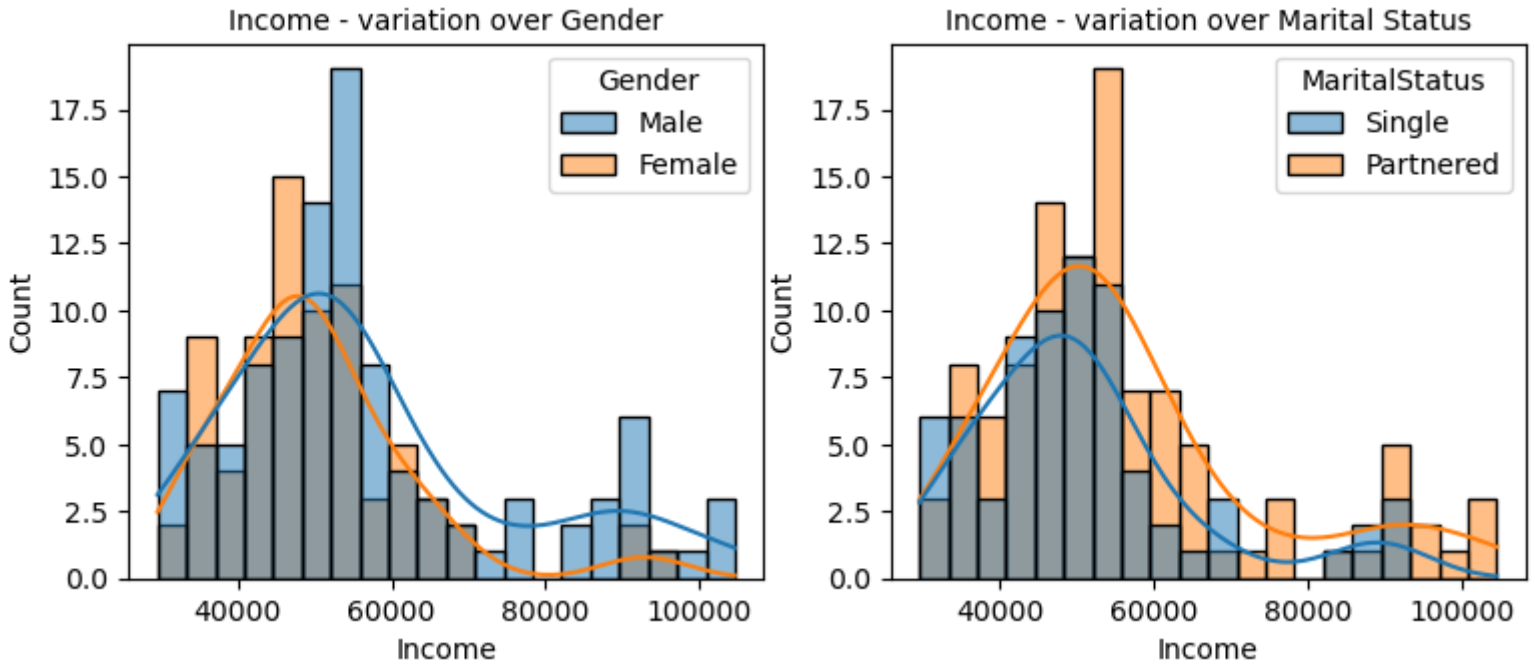
1. Bivariate Analysis: Variation of Fields over Gender and Marital Status field

In [25]:
```
#Graphs/Plots for distribution of product over Gender and Marital Status
plt.figure(figsize = (9,4))
plt.subplot(1,2,1)
sns.countplot(data = data, x = "Gender", hue ="Product" )
plt.title("Distribution of product over Gender", pad=10, fontsize=10)
plt.subplot(1,2,2)
sns.countplot(data = data, x = "MaritalStatus", hue = "Product")
plt.title("Distribution of product over Marital Status", pad=10, fontsize=10)
plt.show()

# Males and females both are equally likely to use product KP281.
# The above trend can be seen for product KP481 also with a little bit less count in females.
# In product KP781, males are more likely to use in compare with females.

# In all products, customers with partners are more active in using products, in comparison
# with singles.
```

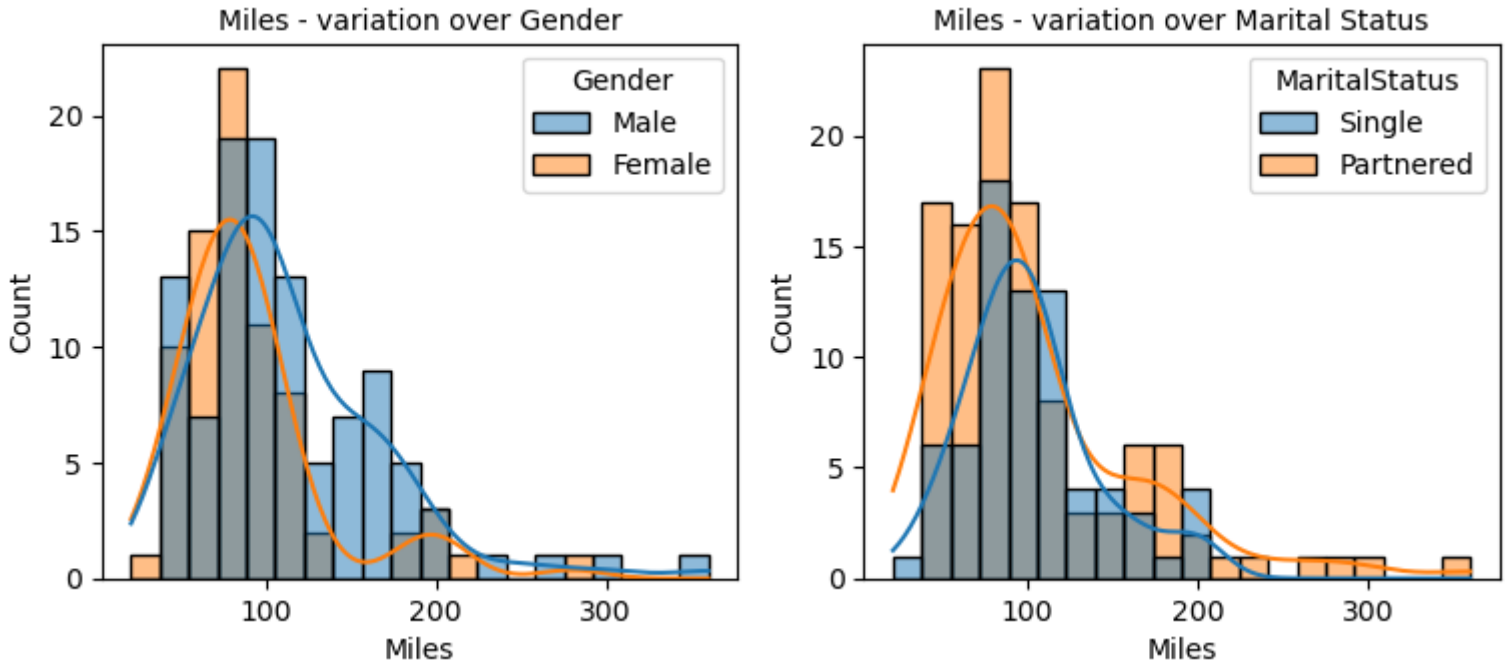

Distribution of product over Gender
Product
KP281
KP481
KP781
count
0
5
10
15
20
25
30
35
40
Male
Female
Gender


Distribution of product over Marital Status
Product
KP281
KP481
KP781
count
0
10
20
30
40
50
Single
Partnered
MaritalStatus

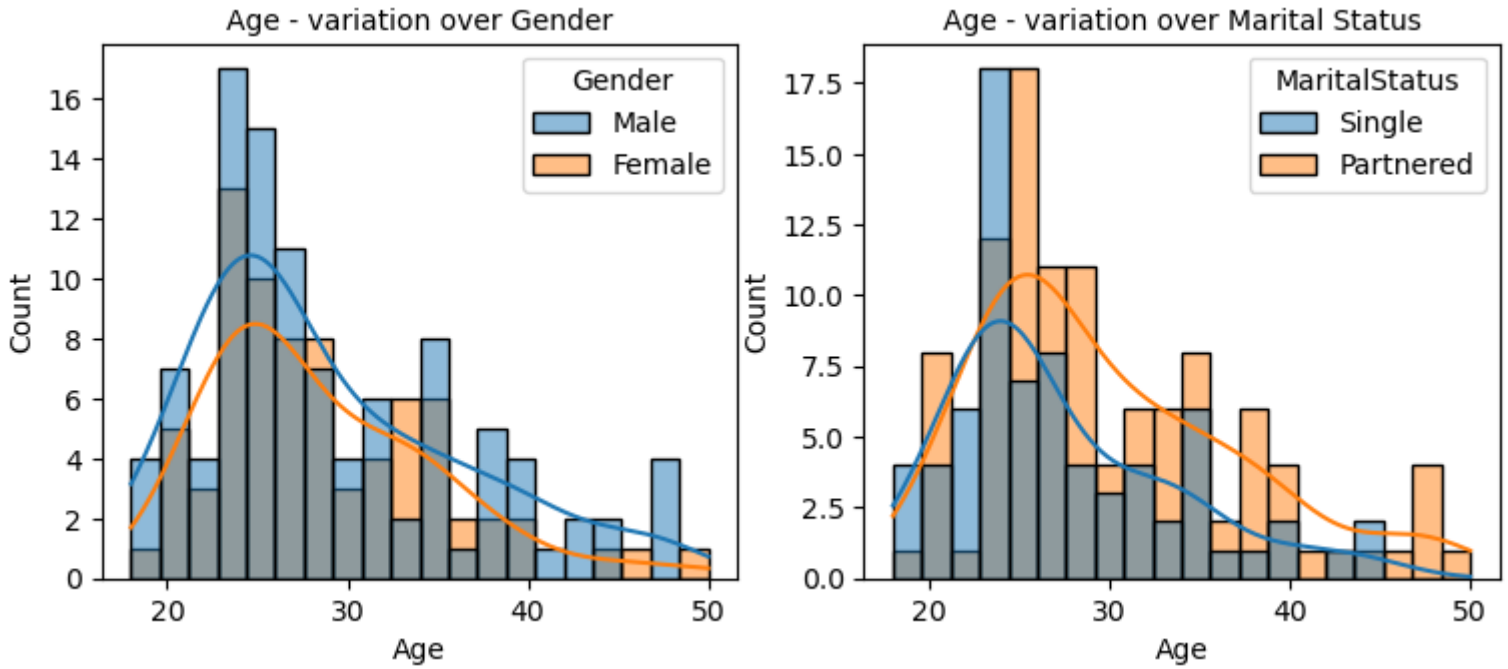
In [87]:
```
#Graphs/Plots for distribution of Income,Miles, and Age over Gender and Marital Status
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.histplot(x='Income',kde='True',data=data, hue='Gender',bins = 20,ax=axis[0])
axis[0].set_title("Income - variation over Gender", pad=5, fontsize=10)
sns.histplot(x='Income',kde='True',data=data, hue='MaritalStatus',bins = 20, ax=axis[1])
axis[1].set_title("Income - variation over Marital Status", pad=5, fontsize=10)
plt.show()
```



In [88]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.histplot(x='Miles',kde='True',data=data, hue='Gender',bins = 20, ax=axis[0])
axis[0].set_title("Miles - variation over Gender", pad=5, fontsize=10)
sns.histplot(x='Miles',kde='True',data=data, hue='MaritalStatus',bins = 20,ax=axis[1])
axis[1].set_title("Miles - variation over Marital Status", pad=5, fontsize=10)
plt.show()
```



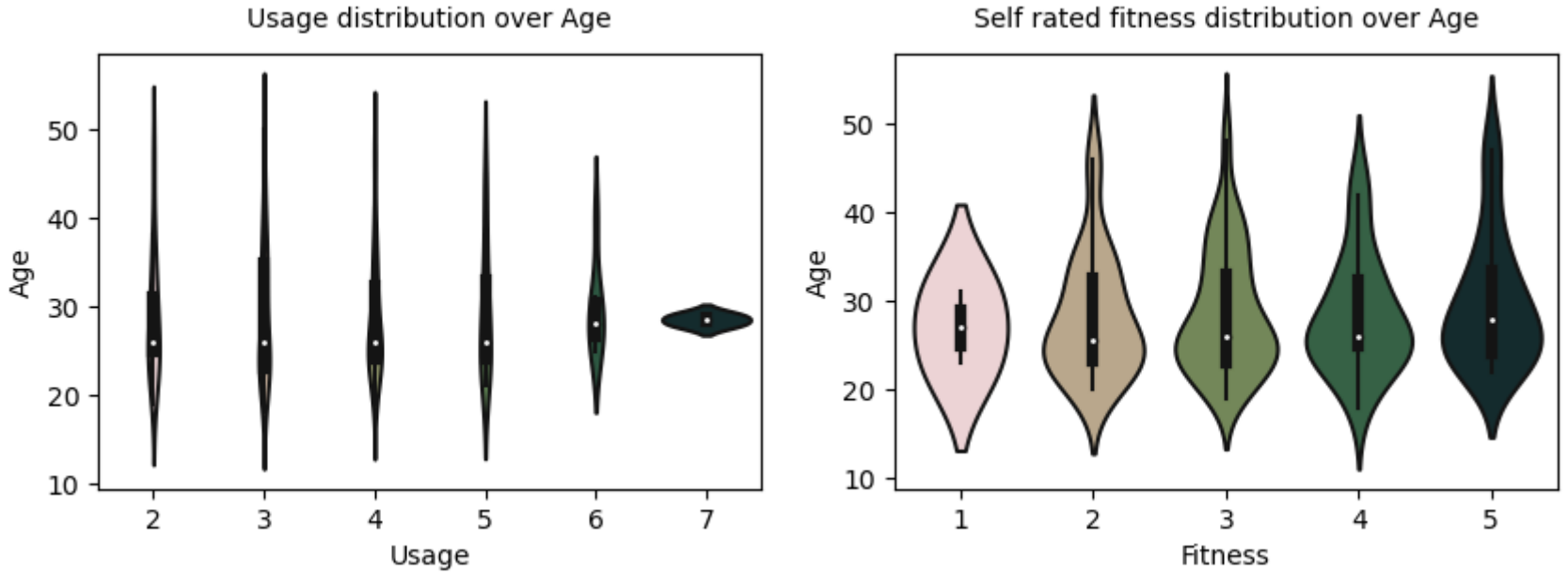
In [89]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.histplot(x='Age',kde='True',data=data, hue='Gender',bins = 20, ax=axis[0])
axis[0].set_title("Age - variation over Gender", pad=5, fontsize=10)
sns.histplot(x='Age',kde='True',data=data, hue='MaritalStatus',bins = 20,ax=axis[1])
axis[1].set_title("Age - variation over Marital Status", pad=5, fontsize=10)
plt.show()
```



1. Male and Female's income are highest in bracket 40k-60k.
2. Males income is bit more than Females over entire range.
3. Partner customers have more income than single customers in all brackets of income.
4. It can be seen above that partner customers use product more than single customers. Income can be one of reason for this.
5. Most Males and females cover 50-120 miles. A drop in females curve can be seen early.
6. Partner customers are more likely to cover more miles compare with single customers.
7. In any bracket of age, the count of males is higher than count of females.
8. Similarly, after 23 years of age, count of partner customers are higher than single customers.

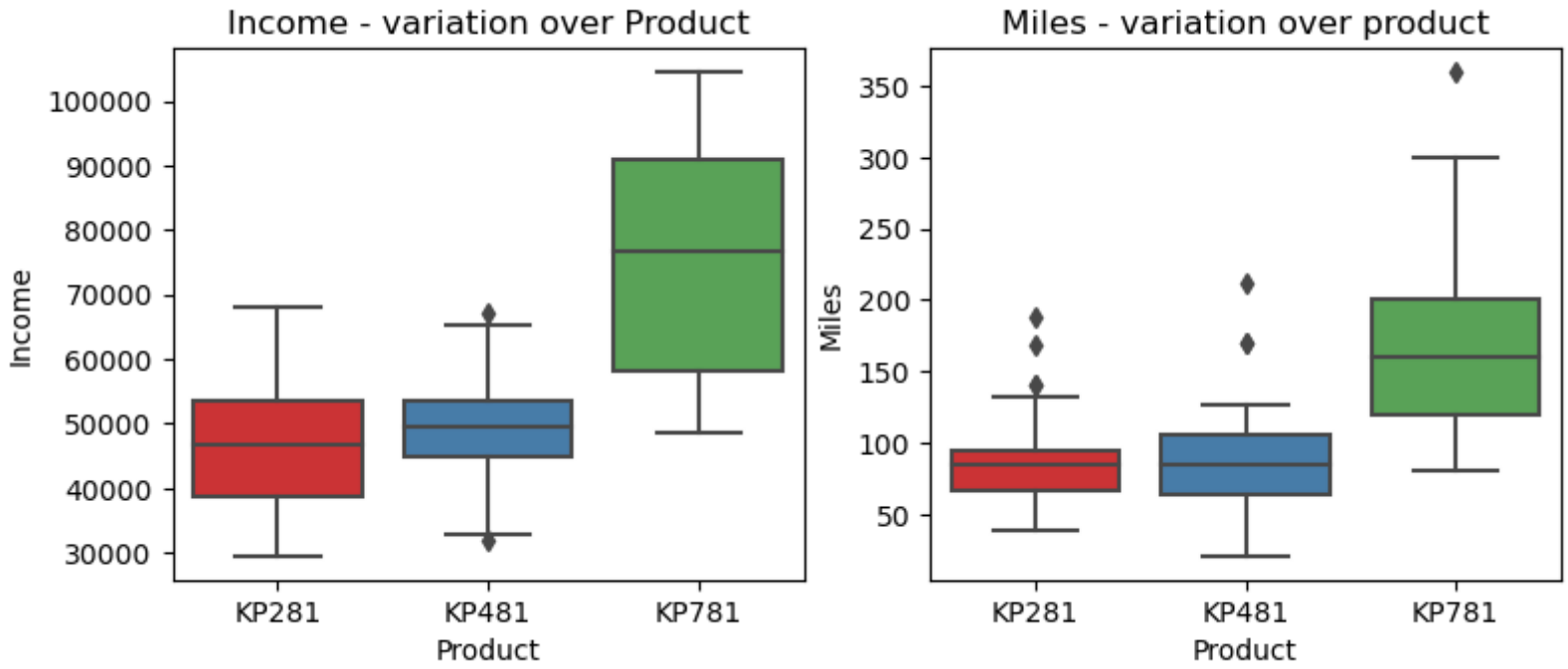
2. Bivariate Analysis: Variation of Fields over Age

In [27]:
```
plt.figure(figsize = (10,3))
plt.subplot(1,2,1)
sns.violinplot(data = data, x='Usage', y = 'Age',palette='ch:s=-.2,rot=-.75')
plt.title("Usage distribution over Age", pad=10, fontsize=10)
plt.subplot(1,2,2)
sns.violinplot(data = data, x='Fitness', y = 'Age',palette='ch:s=-.2,rot=-.75')
plt.title("Self rated fitness distribution over Age", pad=10, fontsize=10)
plt.show()
```

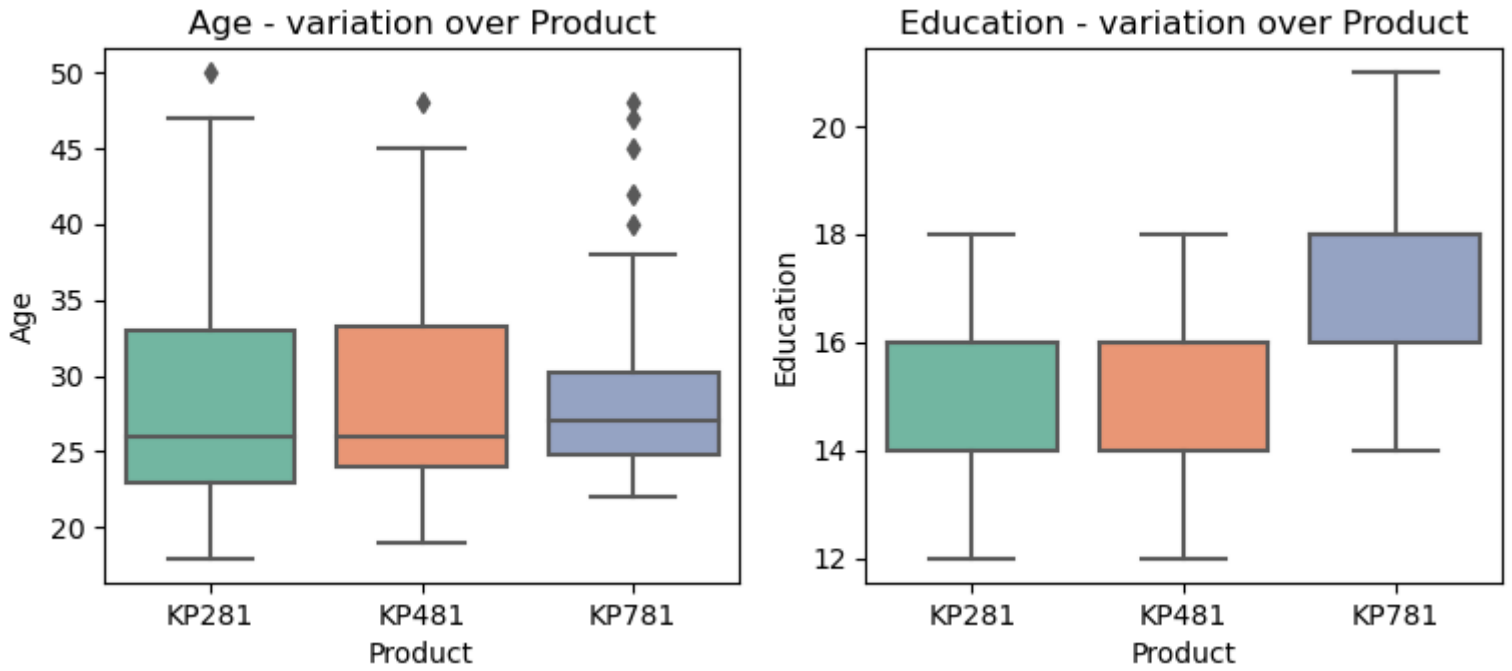


3. Bivariate Analysis: Variation of Fields over types of Product

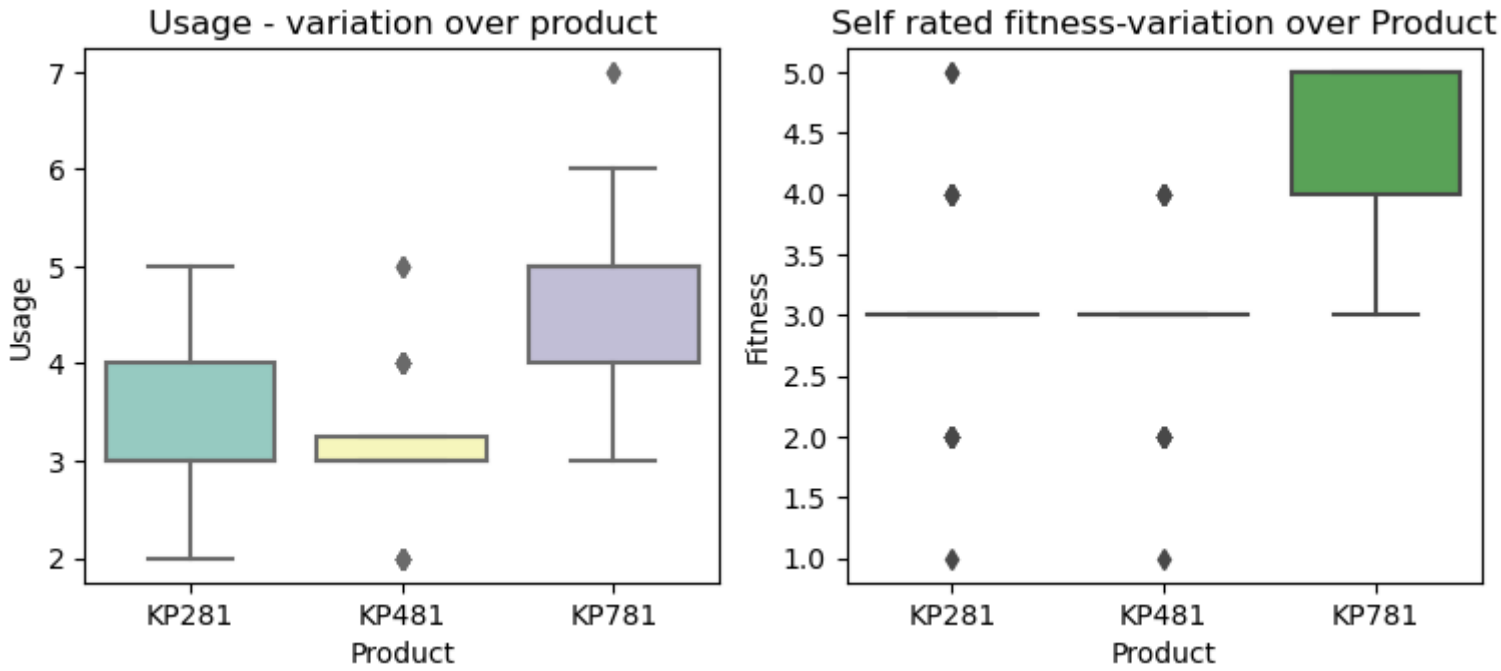
In [106]:
```
#Variation of Income and Miles,Age and Education, Usage and Fitness fields over Product category

fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(x=data["Product"], y = data['Income'], palette='Set1',ax=axis[0])
axis[0].set_title("Income - variation over Product", pad=5, fontsize=12)

sns.boxplot(x=data["Product"], y = data['Miles'], palette='Set1', ax=axis[1])
axis[1].set_title("Miles - variation over product", pad=5, fontsize=12)
plt.show()
```



In [107]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(x=data["Product"], y = data['Age'], palette='Set2', ax=axis[0])
axis[0].set_title("Age - variation over Product", pad=5, fontsize=12)

sns.boxplot(x=data["Product"], y = data['Education'], palette='Set2',ax=axis[1])
axis[1].set_title("Education - variation over Product", pad=5, fontsize=12)
plt.show()
```



In [99]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(x=data["Product"], y = data['Usage'], palette='Set3', ax=axis[0])
axis[0].set_title("Usage - variation over product", pad=5, fontsize=12)

sns.boxplot(x=data["Product"], y = data['Fitness'],palette='Set1',ax=axis[1])
axis[1].set_title("Self rated fitness-variation over Product", pad=5, fontsize=12)
plt.show()
```



Income - variation over Product

1. The mean income of customers using product KP281 and KP481 is very less compare to customers using product KP781.
2. There is one customer who has extremely high income but still using product KP781.

Miles - variation over product

1. Customers using product KP781 are more likely to cover more miles compare to other two products.
2. There are exception customers in all category of products who are covering more than mean miles of their respective product.

Age - variation over Product

1. It can be observed that the mean age of customers using all three kind of products are almost same. However, mean age of customers using product KP781 is little bit more than remaining two.

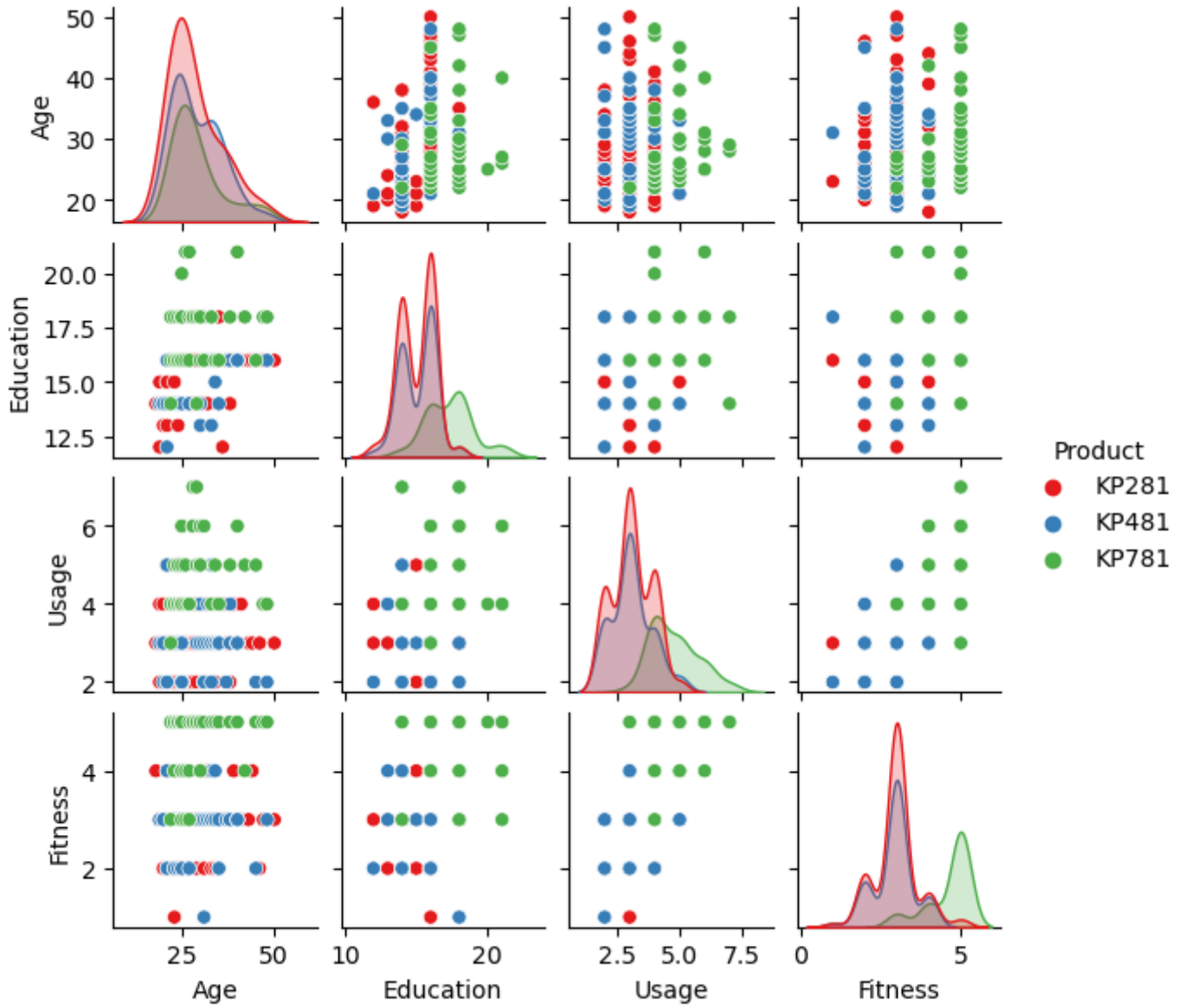
2. There are five outliers in product category KP781, in compare with other two, who have only one outlier.
3. 75% of customers who are using product KP281 are below 33 years. Same result has been observed for product KP481.
4. 75% of customers who are using product KP781 are below 30 years.

Education - variation over Product
1. Customers using product KP281 and KP481 have spend mostly 14 to 16 years in education.
2. Customers using product KP781 have spend mostly 16 to 18 years in education.
3. There are no ouliers in this distribution.

Usage - variation over product
1. For product KP281, maximum customers are using it 3 to 4 times a week.
2. For product KP481, all customers except three, are using it 3 times a week. There are three outliers in this product category.
3. For product KP781, maximum customers are using it 4 to 5 times a week. However, there is one customer who is using it whole week. The minimum days, a customer has used this product is 3 days a week

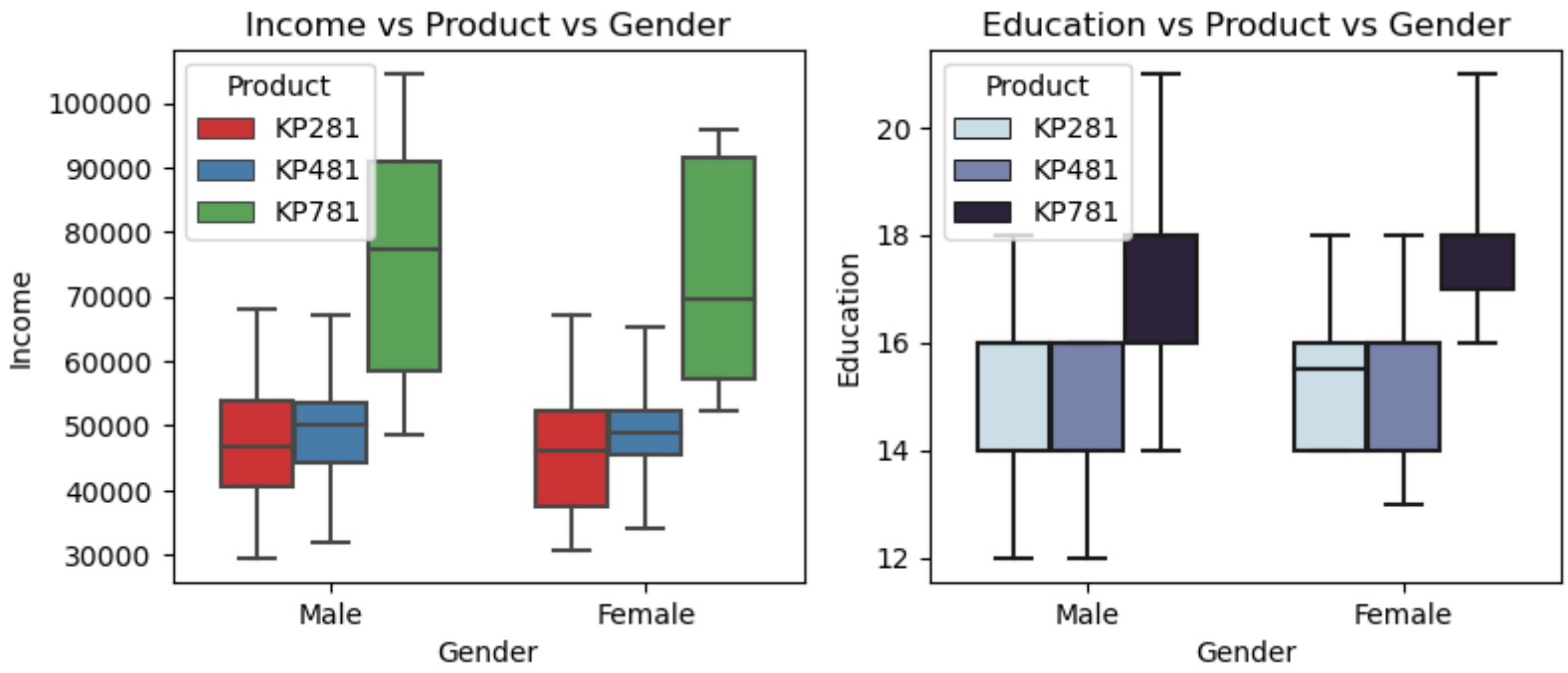
Self rated fitness-variation over Product
1. Customers who have rated their fitness as 3 are more likely to use product KP281 and KP481.
   However, there are four outliers for product KP281. There are four people who have rated themselves in category other than 3 also.
   And for product KP481,there are three outliers. There are three people who have rated themselves as 1, 2 and 4 also.
2. Customers who have rated their fitness as 4 and 5 are more likely to use product KP781. The minimum rating that this product has recorded is 3.

In [128]:
```
#Relation of all fields over Product category
data_product = data[["Product","Age", "Education", "Usage", "Fitness"]]
sns.pairplot(data = data_product, hue = "Product",palette='Set1',height=1.5, aspect=1)
plt.show()
```

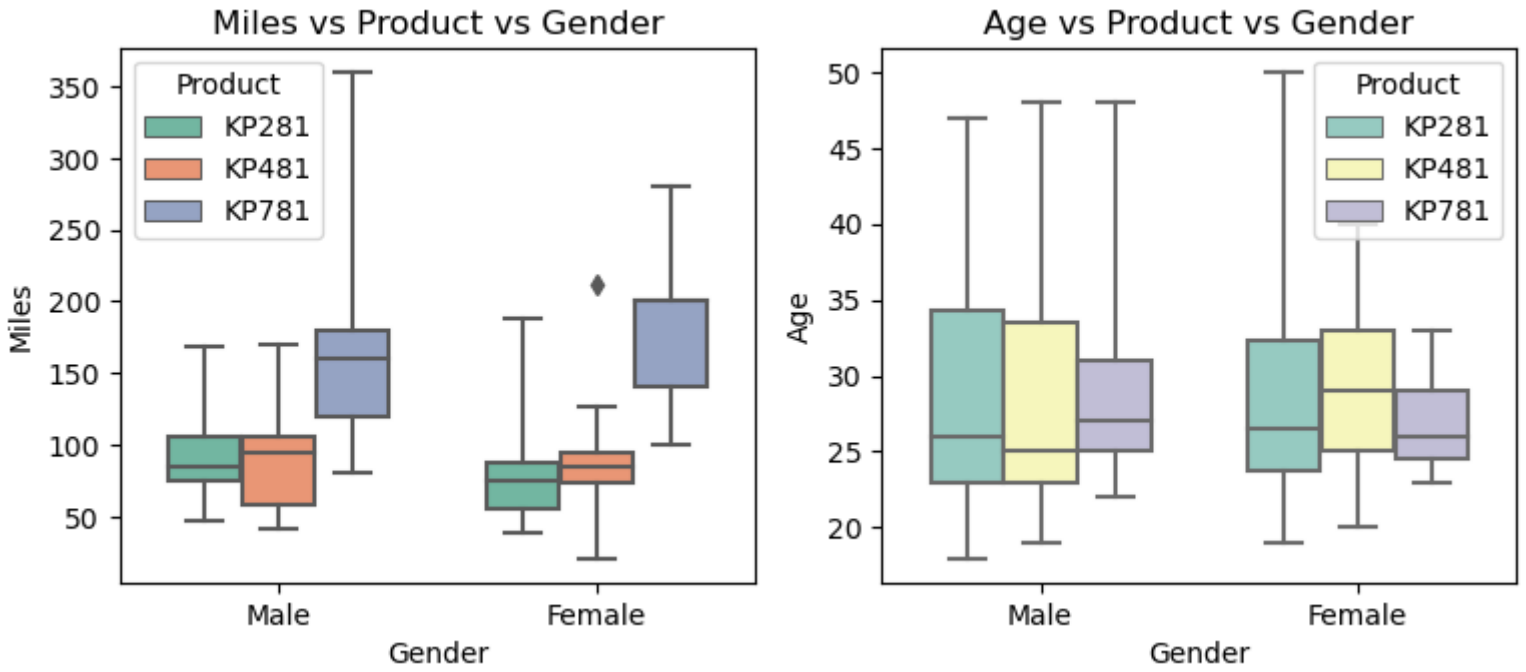


MULTIVARIATE ANALYSIS

In [108]:
```
# Variation of fields for Product vs Gender

fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(data=data, x='Gender', y='Income',palette='Set1',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True,ax=axis[0])
axis[0].set_title("Income vs Product vs Gender", pad=5, fontsize=12)
sns.boxplot(data=data, x='Gender', y='Education',palette='ch:s=.25,rot=-.25',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True, ax=axis[1])
axis[1].set_title("Education vs Product vs Gender", pad=5, fontsize=12)
plt.show()
```



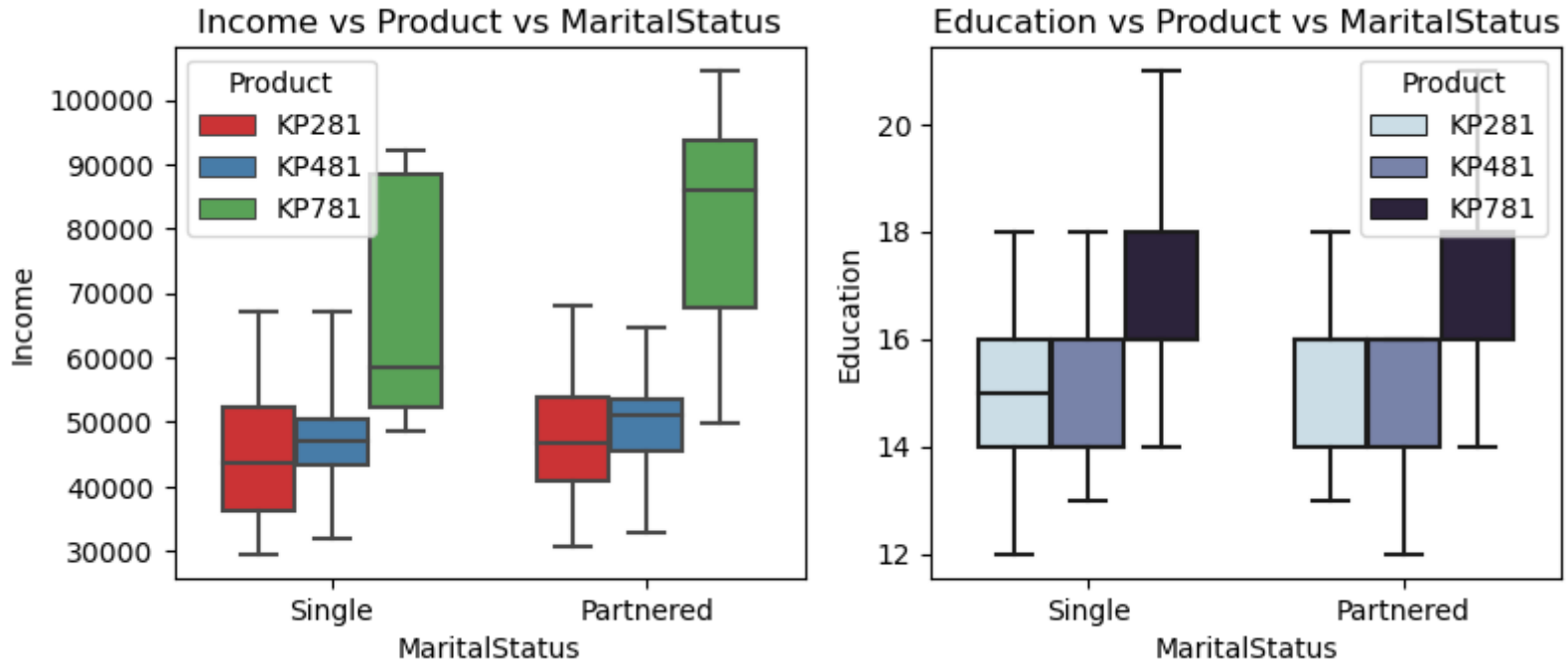
In [109]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(data=data, x='Gender', y='Miles',palette='Set2',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True, ax=axis[0])
axis[0].set_title("Miles vs Product vs Gender", pad=5, fontsize=12)
sns.boxplot(data=data, x='Gender', y='Age',palette='Set3',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True,ax=axis[1])
axis[1].set_title("Age vs Product vs Gender", pad=5, fontsize=12)
plt.show()
```



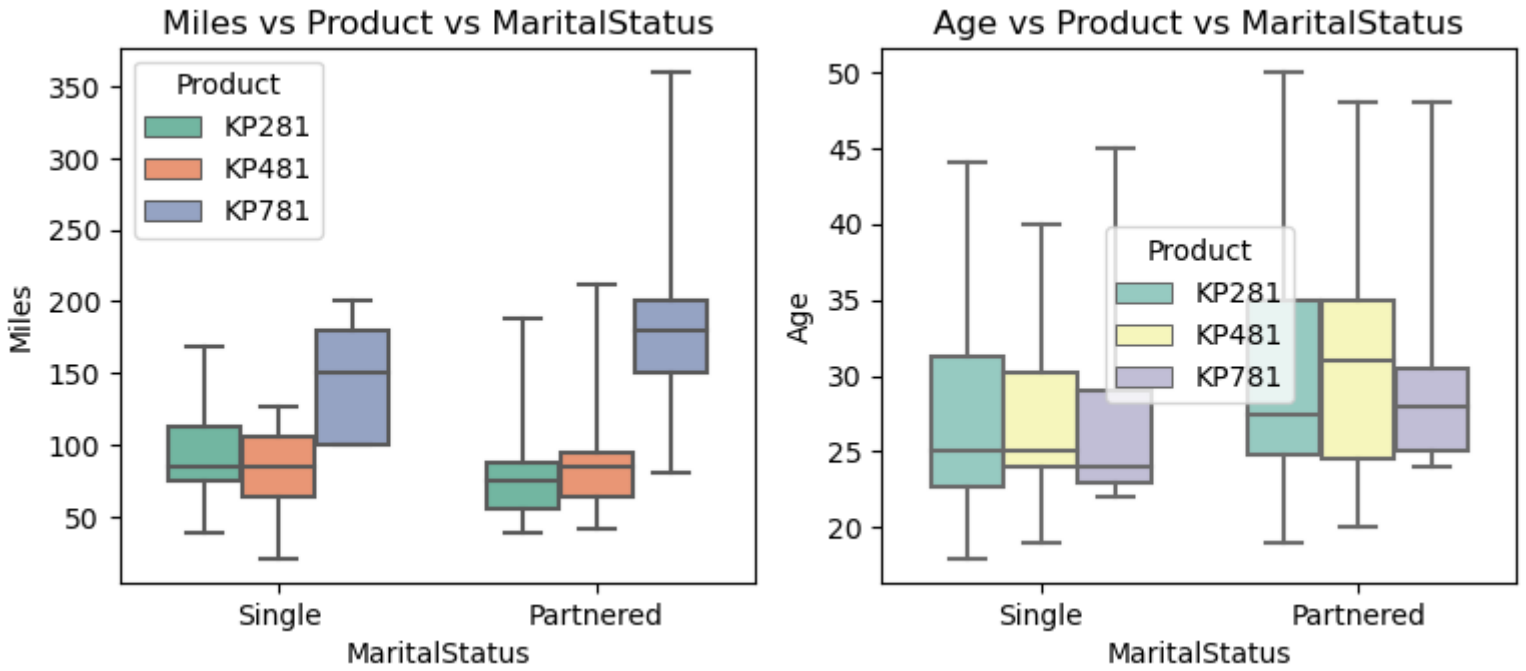
Above plots are showing relation between Product and Gender for various fields.

1. It can be observed that for Income, Education and Miles field, irrespective of what Gender is, the values for product KP281, KP481 are same and value for product KP781 is bit higher than other products.
2. For Age field, reverse can be seen, the age group of customers using product KP781 is less than other product category, irrespective of Gender.

In [110]:
```
# Variation of fields for Product vs Marital Status

fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)

sns.boxplot(data=data, x='MaritalStatus', y='Income',palette='Set1',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True,ax=axis[0])
axis[0].set_title("Income vs Product vs MaritalStatus", pad=5, fontsize=12)
sns.boxplot(data=data, x='MaritalStatus', y='Education',palette='ch:s=.25,rot=-.25',
            hue='Product',width=0.7,whis=5, notch=False, showcaps=True, ax=axis[1])
axis[1].set_title("Education vs Product vs MaritalStatus", pad=5, fontsize=12)
plt.show()
```



In [111]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1)
sns.boxplot(data=data, x='MaritalStatus', y='Miles',palette='Set2',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True, ax=axis[0])
axis[0].set_title("Miles vs Product vs MaritalStatus", pad=5, fontsize=12)
sns.boxplot(data=data, x='MaritalStatus', y='Age',palette='Set3',hue='Product',
            width=0.7,whis=5, notch=False, showcaps=True,ax=axis[1])
axis[1].set_title("Age vs Product vs MaritalStatus", pad=5, fontsize=12)
plt.show()
```



Above plots are showing relation between Product and Marital Status for various fields

1. It can be observed that for Income, Education and Miles field, irrespective of what marital status is, the values for product KP281, KP481 are same and value for product KP781 is bit higher than other products.
2. For Age field, reverse can be seen, the age group of customers using product KP781 is less than other product category, irrespective of what marital status is.
   However, the total usage of KP781 is far more in partner category than single category.

Customer Profile for each AeroFit treadmill

Analysis for product KP281

```
    The KP281 is an entry-level treadmill that sells for 1,500 dollars.
```

In [32]:
```
data_KP281 = data[data["Product"] == "KP281"]

#Probability of customers using product KP281\
p_KP281 = len(data_KP281)/len(data)
print("P(KP281):",round(p_KP281,3))

# The probability of using Product KP281 is 0.44.
```

```
P(KP281): 0.444
```

1. Given Product used is KP281, Probability of customers being Male and Female

In [33]:
```
# 1. Given Product used is KP281, Probability of customers being Male and Female
prob_male_KP281 = len(data_KP281[data_KP281["Gender"] == "Male"])/len(data_KP281)
prob_female_KP281 = len(data_KP281[data_KP281["Gender"] == "Female"])/len(data_KP281)
print(" P(male|KP281):",prob_male_KP281,
      "\n","P(female|KP281):",prob_female_KP281)

# Both Males and females probability of using product KP281 is same.
# Given product used is KP281, probability of being male and female is 0.5 respectively.
```

```
 P(male|KP281): 0.5
 P(female|KP281): 0.5
```

2. Given Product used is KP281, Probability of customers being partnered and Single

In [34]:
```
# 2. Given Product used is KP281, Probability of customers being partnered and Single
prob_partnered_KP281 = len(data_KP281[data_KP281["MaritalStatus"] == "Partnered"])/len(data_KP281)
prob_single_KP281 = len(data_KP281[data_KP281["MaritalStatus"] == "Single"])/len(data_KP281)
print(" P(partner|KP281):",prob_partnered_KP281,
      "\n","P(single|KP281):",prob_single_KP281)

# Partnered customers are using this product more than single customers.
# The probability of customers having partner, given product used is KP281, is 0.6.
# The probability of single customers, given product used is KP281, is 0.4.
```

```
P(partner|KP281): 0.6
P(single|KP281): 0.4
```

3. Given product is KP281, analysis of all numeric data

In [35]:
```
data_KP281.describe()
```

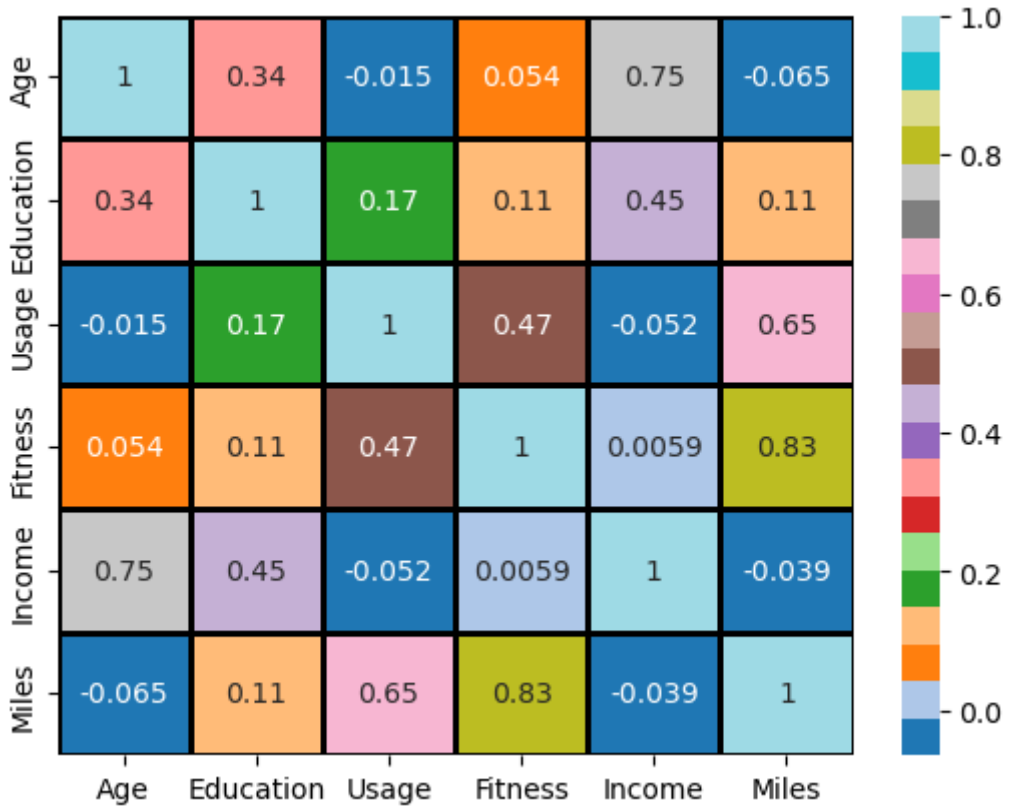
Out[35]:

	Age	Education	Usage	Fitness	Income	Miles
count	80.000000	80.000000	80.000000	80.00000	80.00000	80.000000
mean	28.550000	15.037500	3.087500	2.96250	46418.02500	82.787500
std	7.221452	1.216383	0.782624	0.66454	9075.78319	28.874102
min	18.000000	12.000000	2.000000	1.00000	29562.00000	38.000000
25%	23.000000	14.000000	3.000000	3.00000	38658.00000	66.000000
50%	26.000000	16.000000	3.000000	3.00000	46617.00000	85.000000
75%	33.000000	16.000000	4.000000	3.00000	53439.00000	94.000000
max	50.000000	18.000000	5.000000	5.00000	68220.00000	188.000000

1. The minimum and maximum age of customers using product KP281 are 18 and 50 years respectively.
2. The mean age of customers using this product is around 28.55 years.
3. The maximum miles that have been recorded for this product is 188. However, 75% of customers have covered only 94 miles. It can be infer that there can be outlier in this category.
4. The minimum rating is 1 and maximum rating is 5, But the mean is 2.96 and 75% of customers have rated themselves as 3. It can be concluded that there will be much outliers in this category.

4. Correlation of all fields given product is KP281

In [36]:
```
sns.heatmap(data_KP281.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.show()

# Visual representation of correlation of fields given product used is KP281.
```



Analysis for product KP481

```
    The KP481 is for mid-level runners that sell for 1,750 dollars.
```

In [37]:
```
data_KP481 = data[data["Product"] == "KP481"]

#Probability of customers using product KP481\
p_KP481 = len(data_KP481)/len(data)
print("P(KP481):",round(p_KP481,3))

# The probability of using Product KP281 is 0.33.
```

```
P(KP481): 0.333
```

1. Given Product used is KP481, Probability of customers being Male and Female

In [38]:
```
# 1. Given Product used is KP481, Probability of customers being Male and Female
prob_male_KP481 = len(data_KP481[data_KP481["Gender"] == "Male"])/len(data_KP481)
prob_female_KP481 = len(data_KP481[data_KP481["Gender"] == "Female"])/len(data_KP481)
print(" P(male|KP481):",round(prob_male_KP481,2),
      "\n","P(female|KP481):",round(prob_female_KP481,2))

# Probability of Males using product KP481 is bit higher than probability of females using this product.
# Given product used is KP481, probability of being male is 0.52, whereas being female is 0.48.
```

```
P(male|KP481): 0.52
P(female|KP481): 0.48
```

2. Given Product used is KP481, Probability of customers being partnered and Single

In [39]:
```
# 2. Given Product used is KP481, Probability of customers being partnered and Single
prob_partnered_KP481 = len(data_KP481[data_KP481["MaritalStatus"] == "Partnered"])/len(data_KP481)
prob_single_KP481 = len(data_KP481[data_KP481["MaritalStatus"] == "Single"])/len(data_KP481)
print(" P(partner|KP481):",prob_partnered_KP481,
      "\n","P(single|KP481):",prob_single_KP481)

# Partnered customers are using this product more than single customers.
# The probability of customers having partner, given product used is KP481, is 0.6.
# The probability of single customers, given product used is KP481, is 0.4.
```

```
P(partner|KP481): 0.6
P(single|KP481): 0.4
```

3. Given product is KP481, analysis of all numeric data

In [40]:
```
data_KP481.describe()
```

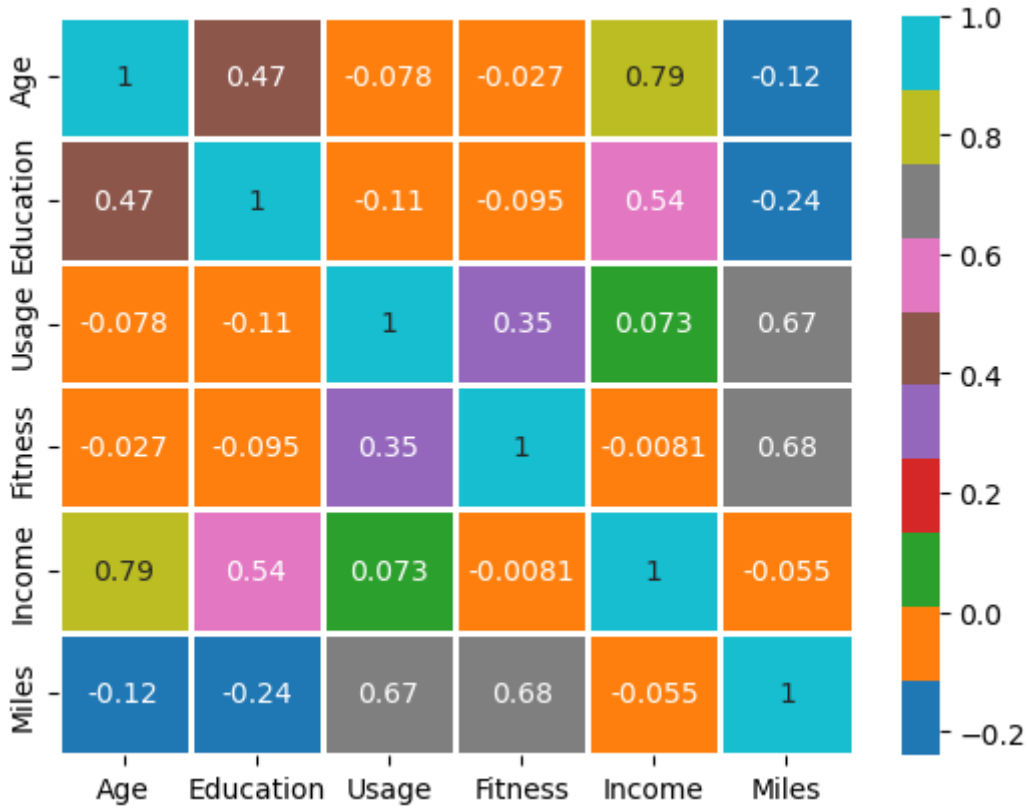
Out[40]:

	Age	Education	Usage	Fitness	Income	Miles
count	60.000000	60.000000	60.000000	60.00000	60.000000	60.000000
mean	28.900000	15.116667	3.066667	2.90000	48973.650000	87.933333
std	6.645248	1.222552	0.799717	0.62977	8653.989388	33.263135
min	19.000000	12.000000	2.000000	1.00000	31836.000000	21.000000
25%	24.000000	14.000000	3.000000	3.00000	44911.500000	64.000000
50%	26.000000	16.000000	3.000000	3.00000	49459.500000	85.000000
75%	33.250000	16.000000	3.250000	3.00000	53439.000000	106.000000
max	48.000000	18.000000	5.000000	4.00000	67083.000000	212.000000

1. The minimum and maximum age of customers using product KP481 are 19 and 48 years respectively.
   The mean age of customers using this product is around 28.90 years.
2. The minimum fitness rating is 1 and maximum rating is 4, But the mean is 2.9 and 75% of customers have
   rated themselves as 3. It can be concluded that there will be much outliers in this category.
3. The maximum miles that have been recorded for this product is 212. However, 75% of customers
   have covered only 106 miles. It can be infer that there can be outlier in this category also.
5. The minimum and maximum usage of KP481 product are 2 days and 5 days a week respectively. And mean is
   3 days a week. It can be infer that there will be outlier in this category.

4. Correlation of all fields given product is KP481

In [41]:
```
sns.heatmap(data_KP481.corr(),linewidths=1, linecolor= 'white', annot = True,
            cbar=True, cmap = "tab10")
plt.show()

# Visual representation of correlation of fields given product used is KP481.
```



Probability Analysis for product KP781

```
The KP781 treadmill is having advanced features that sell for 2,500 dollars.
```

In [42]:
```
data_KP781 = data[data["Product"] == "KP781"]

#Probability of customers using product KP781\
p_KP781 = len(data_KP781)/len(data)
print("P(KP781):",round(p_KP781,3))

# The probability of using Product KP281 is 0.22.
```

```
P(KP781): 0.222
```

1. Given Product used is KP781, Probability of customers being Male and Female

In [43]:
```
# 1. Given Product used is KP781, Probability of customers being Male and Female
prob_male_KP781 = len(data_KP781[data_KP781["Gender"] == "Male"])/len(data_KP781)
prob_female_KP781 = len(data_KP781[data_KP781["Gender"] == "Female"])/len(data_KP781)
print(" P(male|KP781):",prob_male_KP781,
      "\n","P(female|KP781):",prob_female_KP781)

# Probability of Males using product KP481 is much higher than probability of females using this product.
# Given product used is KP781, probability of being male is 0.825, whereas being female is 0.175.
```

```
P(male|KP781): 0.825
P(female|KP781): 0.175
```

2. Given Product used is KP781, Probability of customers being partnered and Single

In [44]:
```
# 2. Given Product used is KP781, Probability of customers being partnered and Single
prob_partnered_KP781 = len(data_KP781[data_KP781["MaritalStatus"] == "Partnered"])/len(data_KP781)
prob_single_KP781 = len(data_KP781[data_KP781["MaritalStatus"] == "Single"])/len(data_KP781)
print(" P(Partner|KP781):",prob_partnered_KP781,
      "\n","P(Single|KP781):",prob_single_KP781)

# Partnered customers are using this product more than single customers.
# The probability of customers having partner, given product used is KP781, is 0.575.
# The probability of single customers, given product used is KP281, is 0.425.
```

```
P(Partner|KP781): 0.575
P(Single|KP781): 0.425
```

3. Given product is KP781, analysis of all numeric data

In [45]:
```
data_KP781.describe()
```

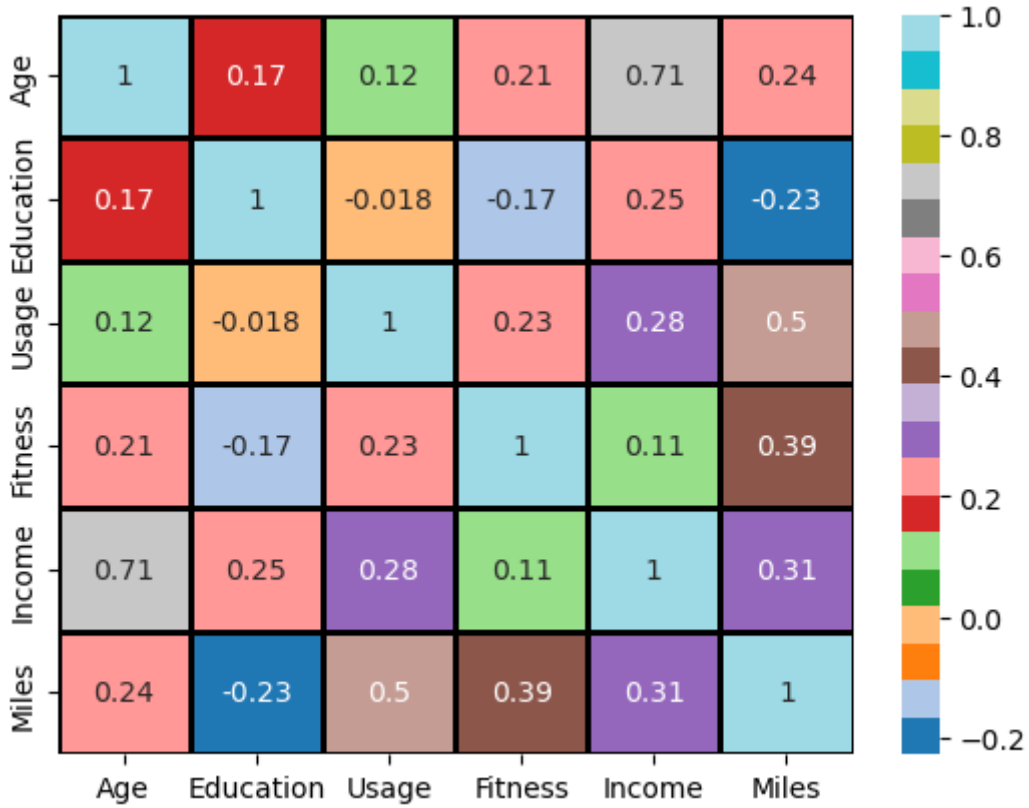
Out[45]:

	Age	Education	Usage	Fitness	Income	Miles
count	40.000000	40.000000	40.000000	40.000000	40.00000	40.000000
mean	29.100000	17.325000	4.775000	4.625000	75441.57500	166.900000
std	6.971738	1.639066	0.946993	0.667467	18505.83672	60.066544
min	22.000000	14.000000	3.000000	3.000000	48556.00000	80.000000
25%	24.750000	16.000000	4.000000	4.000000	58204.75000	120.000000
50%	27.000000	18.000000	5.000000	5.000000	76568.50000	160.000000
75%	30.250000	18.000000	5.000000	5.000000	90886.00000	200.000000
max	48.000000	21.000000	7.000000	5.000000	104581.00000	360.000000

1. The minimum and maximum age of customers using product KP781 are 22 and 48 years respectively. However, the mean age of customers using this product is around 29.10 years and it can be observed that 75% of customers using this product are below 30 years. Hence, it can be concluded that there will be much outliers in this category.
2. The maximum miles that have been recorded for this product is 360. However, major customers have covered 200 miles.

4. Correlation of all fields given product is KP781

In [46]:
```
sns.heatmap(data_KP781.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.show()

# Visual representation of correlation of fields given product used is KP781.
```



Marginal probabilities

In [47]:
```
print("P(KP281):",round(len(data_KP281)/len(data),2))
print("P(KP481):",round(len(data_KP481)/len(data),2))
print("P(KP781):",round(len(data_KP781)/len(data),2))
```

```
P(KP281): 0.44
P(KP481): 0.33
P(KP781): 0.22
```

In [48]:
```
from IPython.display import display, HTML
CSS = """
.output {
    flex-direction: row;
}
"""
HTML('<style>{}</style>'.format(CSS))
```

Out[48]:

In [49]:
```
df1 = pd.crosstab(index= data["Gender"], columns= data["Product"], margins=True, normalize='index')
df2 = pd.crosstab(index= data["MaritalStatus"], columns= data["Product"], margins=True, normalize='index')
display(df1)
display(df2)
```

Product	KP281	KP481	KP781
Gender			
Female	0.526316	0.381579	0.092105
Male	0.384615	0.298077	0.317308
All	0.444444	0.333333	0.222222

Product	KP281	KP481	KP781
MaritalStatus			
Partnered	0.448598	0.336449	0.214953
Single	0.438356	0.328767	0.232877
All	0.444444	0.333333	0.222222

In [50]:
```
df3 = pd.crosstab(index= data["Usage"], columns= data["Product"], margins=True, normalize='index')
df4 = pd.crosstab(index= data["Fitness"], columns= data["Product"], margins=True, normalize='index')
display(df3)
display(df4)
```

Product	KP281	KP481	KP781
Usage			
2	0.575758	0.424242	0.000000
3	0.536232	0.449275	0.014493
4	0.423077	0.230769	0.346154
5	0.117647	0.176471	0.705882
6	0.000000	0.000000	1.000000
7	0.000000	0.000000	1.000000
All	0.444444	0.333333	0.222222

Product	KP281	KP481	KP781
Fitness			
1	0.500000	0.500000	0.000000
2	0.538462	0.461538	0.000000
3	0.556701	0.402062	0.041237
4	0.375000	0.333333	0.291667
5	0.064516	0.000000	0.935484
All	0.444444	0.333333	0.222222

In [51]:
```
df5 = pd.crosstab(index= data["Education"], columns= data["Product"], margins=True, normalize='index')
display(df5)
```

Product	KP281	KP481	KP781
Education			
12	0.666667	0.333333	0.000000
13	0.600000	0.400000	0.000000
14	0.545455	0.418182	0.036364
15	0.800000	0.200000	0.000000
16	0.458824	0.364706	0.176471
18	0.086957	0.086957	0.826087
20	0.000000	0.000000	1.000000
21	0.000000	0.000000	1.000000
All	0.444444	0.333333	0.222222

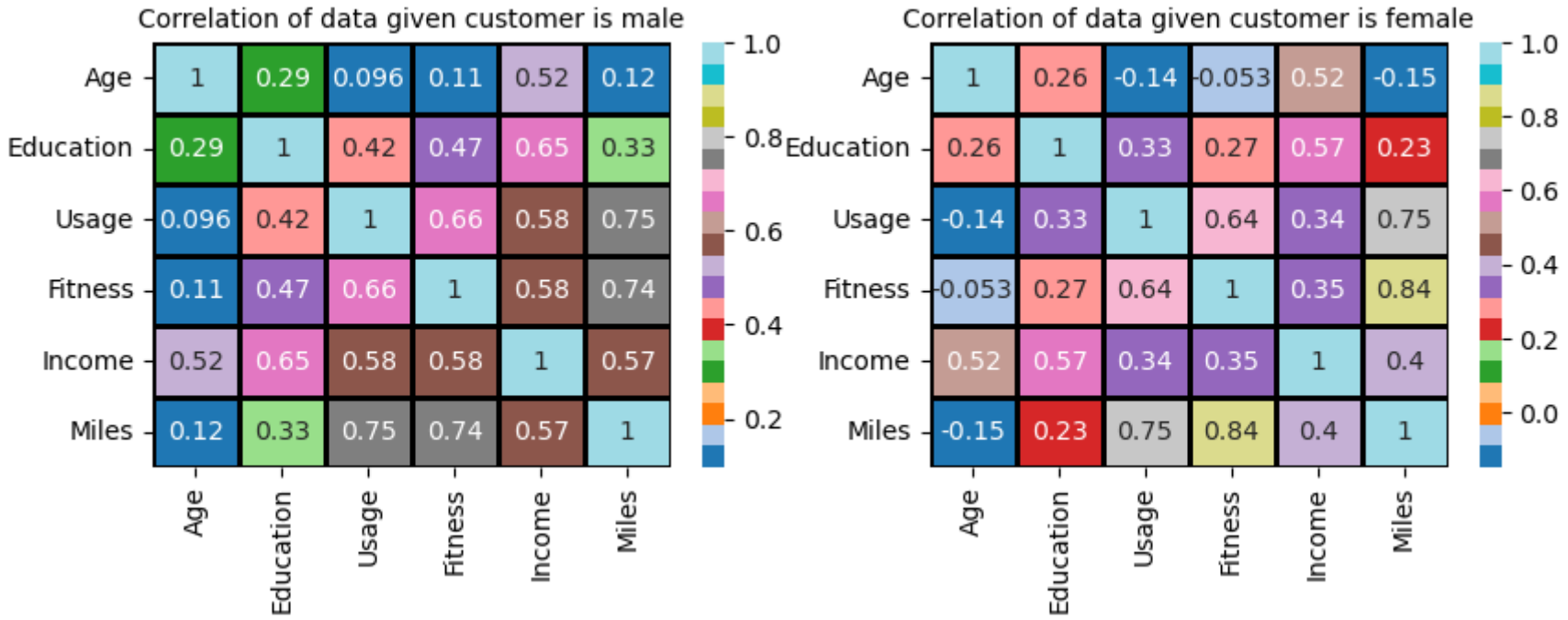
Conditional Probability Analysis for Gender

Given Gender of customers, conditional probability for buying Product Category

In [52]:
```
#data for male and females customers
data_male = data[data["Gender"] == "Male"]
data_female = data[data["Gender"] == "Female"]
p_male = len(data_male)/len(data)
p_female = len(data_female)/len(data)
print(" P(male):",round(p_male,3), "\n","P(female):",round(p_female,3))
print()

#Given Gender is male, probability of buying product KP281, KP481, and KP781

p_male_KP281 = len(data_male[data_male["Product"] == "KP281"])/len(data_male)
p_male_KP481 = len(data_male[data_male["Product"] == "KP481"])/len(data_male)
p_male_KP781 = len(data_male[data_male["Product"] == "KP781"])/len(data_male)
print(" P(KP281|male):",round(p_male_KP281,3), "\n","P(KP481|male):",round(p_male_KP481,3),
     "\n","P(KP781|male):", round(p_male_KP781,3))
print()

#Given Gender is female, probability of buying product KP281, KP481, and KP781

p_female_KP281 = len(data_female[data_female["Product"] == "KP281"])/len(data_female)
p_female_KP481 = len(data_female[data_female["Product"] == "KP481"])/len(data_female)
p_female_KP781 = len(data_female[data_female["Product"] == "KP781"])/len(data_female)
print(" P(KP281|female):",round(p_female_KP281,3),"\n","P(KP481|female):",round(p_female_KP481,3),
     "\n","P(KP781|female):", round(p_female_KP781,3))
```

```
P(male): 0.578
P(female): 0.422

P(KP281|male): 0.385
P(KP481|male): 0.298
P(KP781|male): 0.317

P(KP281|female): 0.526
P(KP481|female): 0.382
P(KP781|female): 0.092
```

In [93]:
```
plt.figure(figsize = (10,3))
plt.subplot(1,2,1)
sns.heatmap(data_male.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.title("Correlation of data given customer is male", fontsize = 10)
plt.subplot(1,2,2)
sns.heatmap(data_female.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.title("Correlation of data given customer is female", fontsize = 10)
plt.show()
```



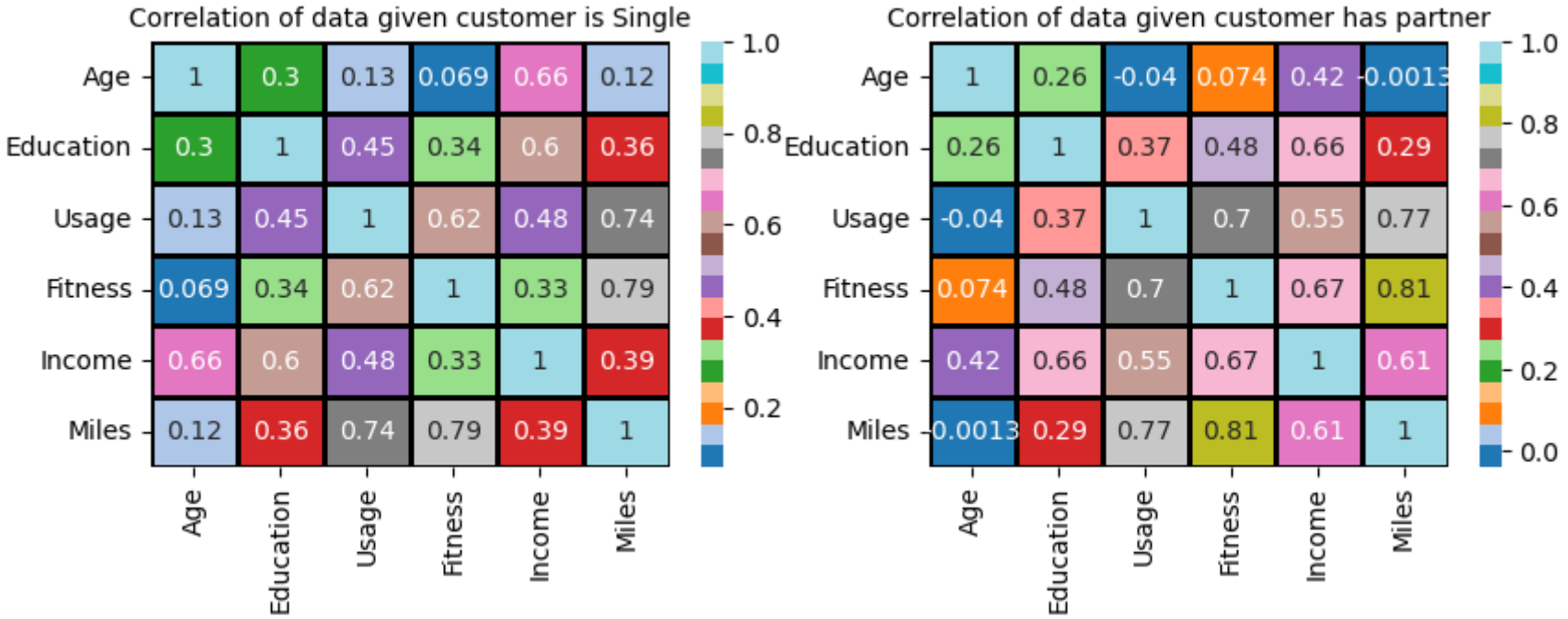
Conditional Probability Analysis for Status of Customers

Given status of customers, conditional probability of buying different Variations of Treadmill

In [54]:
```
#data for single and partnered customers
data_single = data[data["MaritalStatus"] == "Single"]
data_partner = data[data["MaritalStatus"] == "Partnered"]
p_single = len(data_single)/len(data)
p_partner = len(data_partner)/len(data)
print(" P(single):",round(p_single,3), "\n","P(partner):",round(p_partner,3))
print()

#Given Customer is single, probability of buying product KP281, KP481, and KP781
p_single_KP281 = len(data_single[data_single["Product"] == "KP281"])/len(data_single)
p_single_KP481 = len(data_single[data_single["Product"] == "KP481"])/len(data_single)
p_single_KP781 = len(data_single[data_single["Product"] == "KP781"])/len(data_single)
print(" P(KP281|single):",round(p_single_KP281,3), "\n","P(KP481|single):",round(p_single_KP481,3),
     "\n","P(KP781|single):", round(p_single_KP781,3))
print()

#Given Customer has partner, probability of buying product KP281, KP481, and KP781
p_partner_KP281 = len(data_partner[data_partner["Product"] == "KP281"])/len(data_partner)
p_partner_KP481 = len(data_partner[data_partner["Product"] == "KP481"])/len(data_partner)
p_partner_KP781 = len(data_partner[data_partner["Product"] == "KP781"])/len(data_partner)
print(" P(KP281|partner):",round(p_partner_KP281,3),"\n","P(KP481|partner):",round(p_partner_KP481,3),
     "\n","P(KP781|partner):", round(p_partner_KP781,3))
```

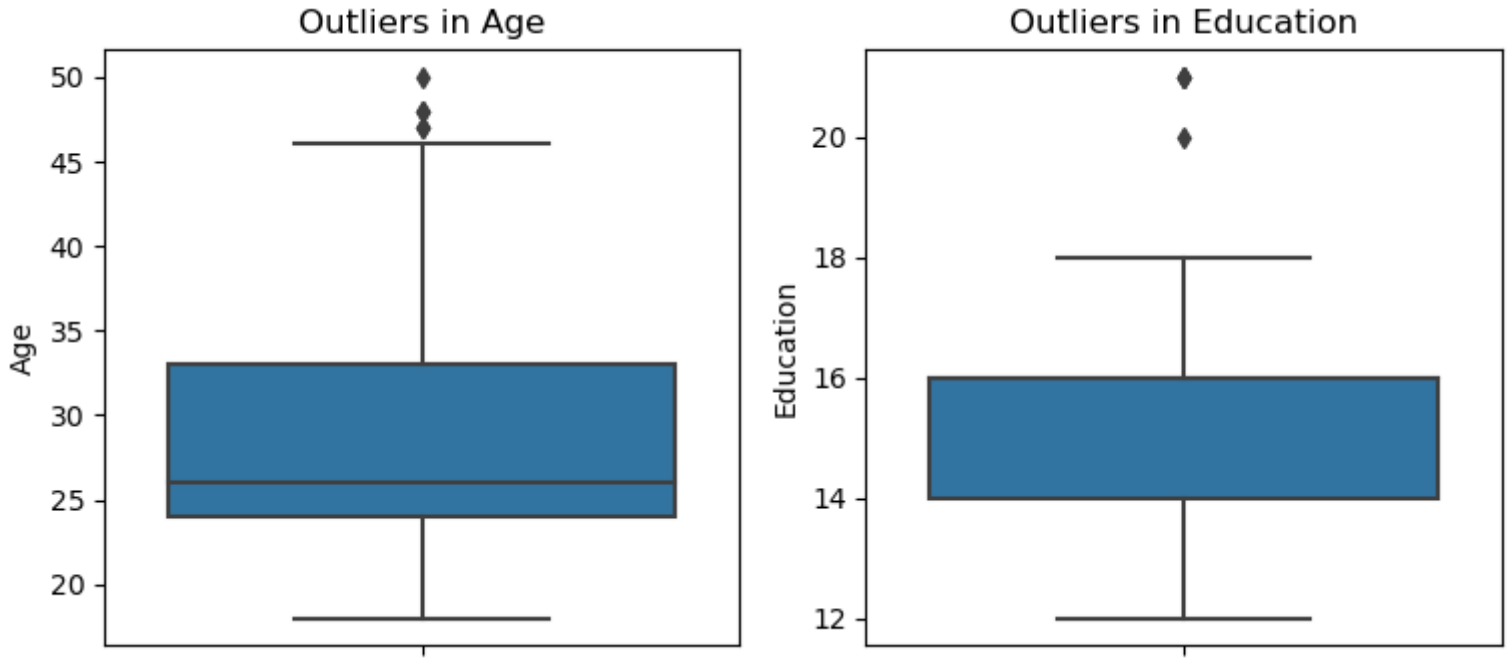
```
P(single): 0.406
P(partner): 0.594

P(KP281|single): 0.438
P(KP481|single): 0.329
P(KP781|single): 0.233

P(KP281|partner): 0.449
P(KP481|partner): 0.336
P(KP781|partner): 0.215
```

In [94]:
```
plt.figure(figsize = (10,3))
plt.subplot(1,2,1)
sns.heatmap(data_single.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.title("Correlation of data given customer is Single", fontsize = 10)
plt.subplot(1,2,2)
sns.heatmap(data_partner.corr(),linewidths=1, linecolor= 'black', annot = True,
            cbar=True, cmap = "tab20")
plt.title("Correlation of data given customer has partner", fontsize = 10)
plt.show()
```

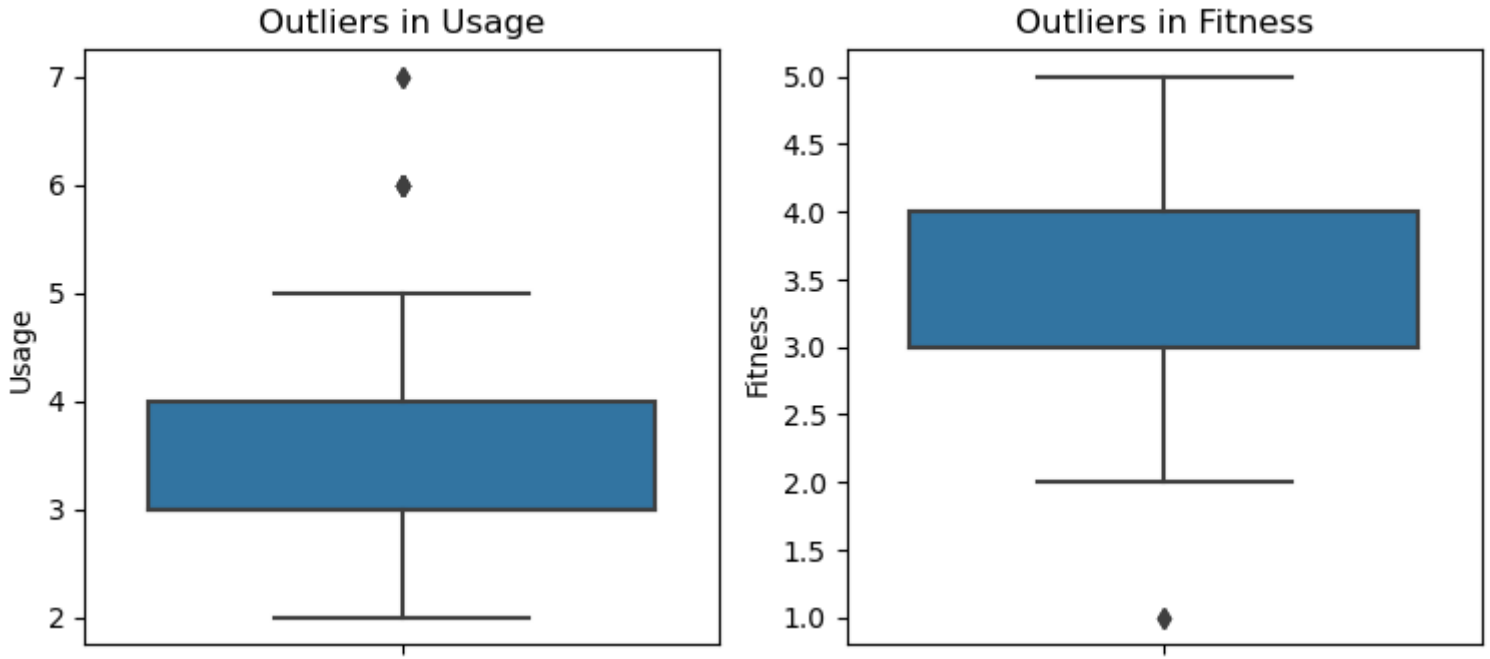


Outliers in dataset

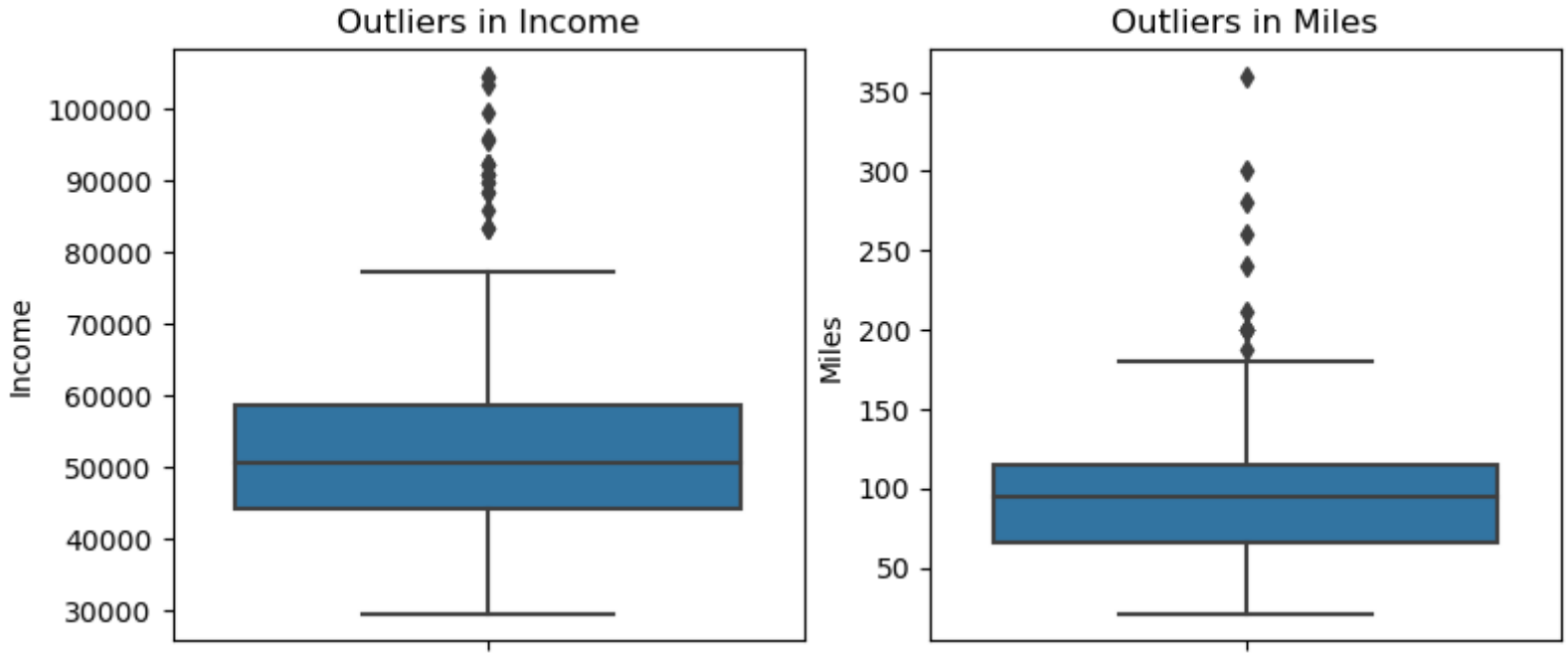
In [91]:
```
# Outliers in Age, education, Usage, Fitness, Income, and Miles
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1.1)

sns.boxplot(y=data["Age"],ax=axis[0])
axis[0].set_title("Outliers in Age")
sns.boxplot(y=data["Education"], ax=axis[1])
axis[1].set_title("Outliers in Education")
plt.show()
```



In [90]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1.1)
sns.boxplot(y=data["Usage"], ax=axis[0])
axis[0].set_title("Outliers in Usage")
sns.boxplot(y=data["Fitness"],ax=axis[1])
axis[1].set_title("Outliers in Fitness")
plt.show()
```



In [92]:
```
fig, axis = plt.subplots(nrows=1, ncols=2, figsize=(9, 3))
fig.subplots_adjust(top=1.1)
sns.boxplot(y=data["Income"], ax=axis[0])
axis[0].set_title("Outliers in Income")
sns.boxplot(y=data["Miles"],ax=axis[1])
axis[1].set_title("Outliers in Miles")
plt.show()
```




```
Outliers in Age:
There are three outliers in Age field. 75% of customers are below 33 years.

Outliers in education:
There are two outliers in Education field. 75% of customers have education experience below 16 years.

Outliers in Usage:
There are only two outliers in Usage field. 75% of customers use treadmill below 4 days a week.

Outliers in Fitness:
There are one outlier in this field.

Outliers in Income:
There are many outliers in the Income field.

Outliers in Miles:
```

```
There are many outliers in the Miles field.
```

Detection and Treating of outliers

```
If deep analysis is not required, less outliers can be ignored.
Fields like Age, Education, Usage and Fitness have very less outliers. Hence, it can be ignored.

Fields like Income and Miles have many outliers. Therefore, treatment of outliers needs to be done.
Outliers can be treated in many ways.
```

In [57]:
```
# Calculating IQR, upper and lower bound for Income in dataset
Q1 = np.percentile(data["Income"], 25,interpolation = 'midpoint')
Q3 = np.percentile(data["Income"], 75,interpolation = 'midpoint')
IQR = Q3 - Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR

print("Upper Bound:",upper)
print("Lower Bound:",lower)

display(data[data["Income"]>=upper][["Product","Income"]])
display(data[data["Income"]<=lower][["Product","Income"]])

# It can be observed that outliers mainly belong to product KP781. This can be concluded that persons
# who are earning more, are using this product.
```

Upper Bound: 81388.25
Lower Bound: 21206.25

	Product	Income
159	KP781	83416
160	KP781	88396
161	KP781	90886
162	KP781	92131
164	KP781	88396
166	KP781	85906
167	KP781	90886
168	KP781	103336
169	KP781	99601
170	KP781	89641
171	KP781	95866
172	KP781	92131
173	KP781	92131
174	KP781	104581
175	KP781	83416
176	KP781	89641
177	KP781	90886
178	KP781	104581
179	KP781	95508

Product	Income

In [58]:
```
# Calculating IQR, upper and lower bound for Miles in dataset
Q1 = np.percentile(data["Miles"], 25,interpolation = 'midpoint')
Q3 = np.percentile(data["Miles"], 75,interpolation = 'midpoint')
IQR = Q3 - Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR

print("Upper Bound:",upper)
print("Lower Bound:",lower)

display(data[data["Miles"]>=upper][["Product","Miles"]])
display(data[data["Miles"]<=lower][["Product","Miles"]])

# It can be observed that outliers mainly belong to product KP781. This can be concluded that persons
# who use this product are covering more miles compare to other product.
```

```
Upper Bound: 192.25
Lower Bound: -9.75
```

	Product	Miles
84	KP481	212
142	KP781	200
148	KP781	200
152	KP781	200
155	KP781	240
166	KP781	300
167	KP781	280
170	KP781	260
171	KP781	200
173	KP781	360
175	KP781	200
176	KP781	200

	Product	Miles

Outliers detection on dataset is not much fruitful as the three products have their different characteristics.

Hence,there is need to detect outliers grouped by three products separately.

In [59]:
```
# Calculating IQR, upper and lower bound for Income in dataset belong to only product KP781
Q1 = np.percentile(data_KP781["Income"], 25,interpolation = 'midpoint')
Q3 = np.percentile(data_KP781["Income"], 75,interpolation = 'midpoint')
IQR = Q3 - Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR

print("Upper Bound:",upper)
print("Lower Bound:",lower)

display(data_KP781[data_KP781["Income"]>=upper][["Product","Income"]])
display(data_KP781[data_KP781["Income"]<=lower][["Product","Income"]])

# For product KP781, there are no outliers in Income field. It can be infer that People with high income
# tend to buy product KP781.
```

```
Upper Bound: 140374.75
Lower Bound: 8404.75
```

Product	Income

Product	Income

In [60]:
```
# Calculating IQR, upper and lower bound for Age in dataset belong to only product KP781
Q1 = np.percentile(data_KP781["Age"], 25,interpolation = 'midpoint')
Q3 = np.percentile(data_KP781["Age"], 75,interpolation = 'midpoint')
IQR = Q3 - Q1

upper=Q3+1.5*IQR
lower=Q1-1.5*IQR

print("Upper Bound:",upper)
print("Lower Bound:",lower)

display(data_KP781[data_KP781["Age"]>=upper][["Product","Age","Income","Miles"]])
display(data_KP781[data_KP781["Age"]<=lower][["Product","Age"]])

# There are five outliers in Age field given product is KP781.
# These ouliers can be deleted from dataset but if we closely observe the below dataset:
# The outliers have minimum and maximum income as 83416 and 104581 respectively.
# These customers become outliers in only Age field. If these data will be deleted, it will affect
# other fields also. Therefore, it is better not to remove them.
```

```
Upper Bound: 39.5
Lower Bound: 15.5
```

	Product	Age	Income	Miles
175	KP781	40	83416	200
176	KP781	42	89641	200
177	KP781	45	90886	160
178	KP781	47	104581	120
179	KP781	48	95508	180

	Product	Age

Business Insights

Expected Revenue

In [67]:
```
# Expected Revenue per product
Expected_revnue_KP281 = round(p_KP281*1500,2) #(Probability of selling KP281)*(KP281 sells or 1500 dollars)
Expected_revnue_KP481 = round(p_KP481*1750,2) #(Probability of selling KP481)*(KP481 sells or 1750 dollars)
Expected_revnue_KP781 = round(p_KP781*2500,2) #(Probability of selling KP781)*(KP781 sells or 2500 dollars)
print("Expected revnue for product KP281:",Expected_revnue_KP281,"dollars" )
print("Expected revnue for product KP481:",Expected_revnue_KP481,"dollars" )
print("Expected revnue for product KP781:",Expected_revnue_KP781,"dollars"  )
print("Expected total revenue is",Expected_revnue_KP281+Expected_revnue_KP481+Expected_revnue_KP781,
     "dollars")

# The total expected revenue of the company is 1805.56 dollars.
# Even though probability of product KP781 is less, but revenue generated by this product is quite
# comparable with other products.
# The expected revenue for product KP281 is highest among revenue generated by other two products.
```

```
Expected revnue for product KP281: 666.67 dollars
Expected revnue for product KP481: 583.33 dollars
Expected revnue for product KP781: 555.56 dollars
Expected total revenue is 1805.56 dollars
```

Customer Profiling - Categorization of users with respect to product

Treadmill Product KP281

```
    This product is best for those customers who have less income, not much fitness enthusiast, and are
    willing to maintain fitness by normal running.

1.  Product KP281 is an entry-level treadmill. It costs around 1,500 dollars.
2.  Around 44% from total customers are using this product.
3.  Men and women are equally using this product. The probability of using this product by males
    and females are equal, that is, 0.5.
4.  Customers having partners are using this product more than single customers. The probability of
    partnered customers is 0.6 whereas, single customers have only 0.4 probability.
5.  This product is used by both young adults and middle-aged adults. However, 75% of customers who
    are using this product is below 33 years.
6.  Customer base of this product hasn't spent much years in education. The maximum year recorded for
    this product is 18 years.
7.  Customers are likely to use this product thrice in a week. Not a single customer use this product
```

whole week.
8. 75% of customers who are using this product do not consider themselves in excellent shape.
9. The annual income of customers using this product is less as compared to customers using other products.
10. Customers using this product covered less miles compared to other products.

Insights about Product KP281:
1. The product KP281 is less costly but lack in good or advance features.
2. They are good products for those customers who have less income and are willing to shape their fitness.
3. Customers who are fitness enthusiast, do not consider this product.
4. The product is equally good for women also.
5. Ideal for partnered customers. The product can be used by both partners and provide nominal fitness shape.
6. As it is not designed for fitness enthusiast, it can be used by middle-aged persons also.

Treadmill Product KP481

This product is best for those customers who have less income, not much fitness enthusiast, but prefer more running and are willing to maintain nominal fitness

1. Product KP481 is for mid-level treadmill. It costs around 1,750 dollars.
2. Around 33% from total customers are using this product.
3. Men are more likely to use this product. The probability of using this product by males is 0.52, whereas probability of females is 0.48.
4. Customers having partners are using this product more than single customers. The probability of partnered customers is 0.6 whereas, single customers have only 0.4 probability.
5. This product is used by both young adults and middle-aged adults. However, 75% of customers who are using this product is below 33 years.
6. Customer base of this product hasn't spent much years in education. The maximum year recorded for this product is 18 years.
7. Customers are likely to use this product thrice in a week. Not a single customer use this product whole week.
8. 75% of customers who are using this product do not consider themselves in excellent shape. Even the maximum rating that customer has self rated is 4.
9. The annual income of customers using this product is less. The mean annual income of customers using this product is 48973.65 dollars which is slightly greater than the mean income of customers using product KP281.
10. Customers using this product covered less miles compared to product KP781.

Insights about Product KP481:
1. The product KP481 is quite costly than product KP281.
2. It has better features than KP281 and good for mid-level runners.
3. Customer base for this product is quite similar to customer base of product KP281.
4. They are better product than KP281 for those customers who have less income and are willing to shape their fitness.

3. Customers who are fitness enthusiast, do not consider this product also.
4. This product is used mostly by men compared to womans.
5. Ideal for partnered customers. The product can be used by both partners and provide nominal fitness shape.

Treadmill Product KP781

This product is best for young customers who are fitness enthusiast, have good income and are willing to maintain good fitness.

1. Product KP781 is a treadmill with advanced features. It costs around 2,500 dollars.
2. Around 22% from total customers are using this product.
3. Mens are highly likely to use this product. The probability of using this product by males is 0.83, whereas probability of females is 0.17 only.
4. A little bit decline can be shown for this product in partner category compare with other products. Customers having partners are using this product more than single customers. The probability of partnered customers is 0.575 whereas, single customers have only 0.425 probability.
5. This product highly used by young adults. 75% of customers who are using this product is below 30 years.
6. Customer base of this product have spent much years in education. The maximum year recorded for this product is 21 years.
7. Customers are likely to use this product five times in a week. Even, some customers are using this product whole week.
8. Maximum customers who are using this product do consider themselves in excellent shape.
9. The annual income of customers using this product is far more compare to customers using other products.
10. Customers using this product covered more miles compare to customers using other products.

Insights about Product KP781:
1. The product KP781 is costly and having good or advance features.
2. They are good products for those customers who have considerably good income and are willing to shape their fitness.
3. Customers who are fitness enthusiast, always opt for this product.
4. The product is highly used by mens.
5. As it is designed with good and advance features, it is generally preferred by young fitness enthusiast.

Recommendations

1. Company needs to make their new customers fill a form which gathers information like age of customer, income of customer, willingness for fitness etc. Based on the gathered information, company can promote their product category in right direction.
2. If the customer is young and earning good and quite a fitness enthusiast, company should recommend product KP781 to them and promote all the features of that product in synchronization with customer's

qualities and desires.
3. No matter whether customer is young or middle-aged, if the customer is earning quite well, company should recommend product KP481.
4. If the customers likes to run, KP481 must be recommended to them.
5. If the customers of any age are earning okay and wants to be in good shape, product KP281 must be recommended to them.
6. For partnered customers, both choice of products KP281 and KP481 are best for recommendation.
7. For Single customers, KP781 is best choice. However, if the customers are not earning well, Company can switch to product KP481.
8. Company should promote product KP781, as it has better features than others and customers using this product are likely to rate their fitness as 5, which in turn helps in selling this product even more.
9. Higher sale of product KP781 will fetch more revenue. Hence, company should promote this product. However, company should not be rigid. If the customer is less enthusiast and not earning good, company should come up with next category KP481.